**Interactive Computation, Parameter Continuation, and Visualization**

John H. Maddocks *, Robert S. Manning, Randy C. Paffenroth,
Kathleen A. Rogers, and Jeremy A. Warner

*Laboratory for Computation and Visualization in Mechanics*
*Institute for Physical Science and Technology*
*and Department of Mathematics*
*University of Maryland*
*College Park, MD 20742*

`http://www.lcvm.umd.edu/`
`jhm@ipst.umd.edu`

November 18, 1996

Submitted to the International Journal of Bifurcation and Chaos

**Abstract**

Nonlinear problems arising in modelling applications are frequently parameter dependent, so that families of solutions are of interest. Such problems naturally lend themselves to interactive computation that exploits parameter continuation methods combined with visualization techniques. Visualization provides both understanding of the solution set and feedback for computational steering. We describe various issues that have arisen in our investigations of problems of this general type.

Keywords: elastic rods, computational steering, bifurcation diagrams, boundary value problems, stability exchange

*Author to whom correspondence should be addressed.

# 1 Introduction

The computation at the heart of many mathematical models of physical systems is the solution of a parameter-dependent system of nonlinear equations of the abstract form

$$\mathbf{g}(\mathbf{u}, \boldsymbol{\lambda}) = \mathbf{0}. \tag{1}$$

Here $\mathbf{u} \in \mathbb{R}^n$ are the basic unknowns, $\boldsymbol{\lambda} \in \mathbb{R}^m$ are parameters, and the function $\mathbf{g}$ maps $\mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ so that there are the same number of equations as unknowns, and the system is formally well-posed when values of the parameters $\boldsymbol{\lambda}$ are prescribed. Within our research group, there is an emphasis on the computation of equilibrium (or other steady state) solutions of dynamical systems, in which case (1) expresses the equilibrium conditions. With this interpretation, there is a natural hierarchy of problem sizes. The simplest examples involve the dynamics of a function of time alone, governed by an ordinary differential equation (ODE) $\mathbf{u}'(t) = \mathbf{g}(\mathbf{u}, \boldsymbol{\lambda})$, and the equilibrium equations are exactly (1). Such problems typically involve fewer than a hundred unknowns. Larger-scale problems arise when one or more spatial variables are added, so that the dynamics involve a partial differential equation (PDE) for a function $\mathbf{U}(x, y, \ldots, t)$ with appropriate spatial boundary conditions. The equilibrium condition $\partial\mathbf{U}/\partial t = \mathbf{0}$ then gives a boundary value problem (BVP) in space, which leads to a system of the form (1) after $\mathbf{U}$ is discretized into unknowns $\mathbf{u} \in \mathbb{R}^n$. With one spatial variable, the equilibrium equations involve a two-point ODE boundary value problem, and the number of unknowns $n$ after spatial discretization is typically on the order of a few thousand. With several spatial variables, the equilibrium equations typically involve an elliptic PDE, and $n$ could be much larger.

The choice of spatial discretization (e.g., finite element, finite difference, collocation) is a crucial consideration in the choice of efficient computational algorithms. However, the focus of this paper is on visualization tools, so we will not discuss problem-specific numerics. Instead, we emphasize techniques for visualization and interactive computation of families of solutions of (1). While interactivity is not a precisely defined concept due to large variations in available dedicated computational resources and the degree of patience of the operator, the computational requirements of interactivity currently limit us to problems with either zero or one spatial dimension. In addition, we only vary $m = 1$ or 2 parameters at a time, since the size of the computations grows exponentially in $m$. We remark that even with only two parameters and moderately sized problems, computational steering is desirable in order to concentrate on interesting regions of parameter space.

Our interests lie in the exploitation of the presence of parameters in the problem for both analysis and computation. In particular, we solve (1) numerically with parameter continuation: solutions at new parameter values are obtained using information about solutions at nearby parameter values. There are now a number of robust and well-tested one-parameter continuation codes, e.g. PITCON [14] and AUTO [2]. (We happen to use the AUTO code extensively in our computations.) One of our observations is that while one strategy for multi-parameter continuation is to perform multiple one-parameter computations, there are circumstances where such a simple approach seems inadequate (see, e.g., Section 5), so that an explicitly multi-dimensional continuation algorithm [6, 13] would be preferable. An extensive survey of available continuation software can be found in [1].

Some special structures of (1) that bear on appropriate visualization techniques arise when the underlying dynamics is a Hamiltonian or a gradient flow. In either case the steady-state equations assume the gradient form

$$G_{\mathbf{u}}(\mathbf{u}, \boldsymbol{\lambda}) = \mathbf{0}. \tag{2}$$

For Hamiltonian dynamical systems the conditions for steady state solutions often specialize further (see [11] for example):

$$F_{\mathbf{u}}(\mathbf{u}, \boldsymbol{\mu}) \equiv H_{\mathbf{u}}(\mathbf{u}) + \sum_i \mu_i C^i_{\mathbf{u}}(\mathbf{u}) = \mathbf{0}. \tag{3}$$

Here $H$ is a Hamiltonian, $F$ is a variational Lagrangian, and the parameters $\lambda_i$ appearing in (2) are now regarded as Lagrange multipliers $\mu_i$ maintaining prescribed values of the functions $C^i(\mathbf{u})$.

In this article we discuss various aspects of visualization that have arisen during the development of two software packages: (i) $PCR$, by Paffenroth and G. Domokos, a post-computational visualization tool that aids in the understanding of solution sets, and (ii) $MC^2$, by Maddocks and C. K. Mesztenyi, a package for the interactive steering of two-parameter continuation computations.

$PCR$ displays various projections of one-parameter bifurcation diagrams and associated solutions. The general implementation of $PCR$ for nonlinear two-point boundary value problems is described in Section 2, and three example applications involving rod mechanics are presented in Sections 3, 4, and 5. The examples illustrate the use of $PCR$ graphics to display problem-specific information, and a natural data compression available for many two-point boundary value problems, namely reconstruction of the full solution from stored initial data. In particular, $PCR$ can be used to interactively visualize different projections of a pre-computed solution set. Certain special projections are particularly useful, such as 'topologically correct' projections, in which branches intersect only if a bifurcation actually occurs [3], or 'distinguished' projections, in which variational structure such as (2) is exploited to reveal maximal stability exchange information [10, 15].

Our discussion of visualization in the interactive steering of computation is centered on the two-parameter continuation package $MC^2$ (an acronym for Multiplier and Constraint Continuation). As described in Section 6, $MC^2$ is optimized for problems of the particular form (3) where stability exchange arises in a very special way that is highly amenable to visualization of certain distinguished bifurcation surfaces [11]. Nevertheless many features of the $MC^2$ paradigm are pertinent to more general two-parameter continuation problems.

## 2 The $PCR$ Visualization Tool

In this section we describe the use of the package $PCR$ [4, 5] as a visualization post-processor for two-point boundary value problems (BVPs) which depend on a single parameter. After discretization, a problem of the form (1) with $m = 1$ is obtained. The solution set, or *bifurcation diagram*, of such a problem is generally a union of one dimensional manifolds, or branches, embedded in $\mathbb{R}^{n+1}$. The data structure used in *PCR* therefore stores the bifurcation diagram as a set of branches, with each branch stored as a one dimensional array of vectors, and with each vector corresponding to a solution of (1). The components of each vector are unknowns $\mathbf{u}$, parameters $\boldsymbol{\lambda}$, and possibly additional related quantities of interest (such as an energy of the solution) that are comparatively expensive to recompute.

Since the bifurcation diagram exists in a high dimensional space, it is necessary to project in order to exploit visualization. To this end, *PCR* provides various three dimensional projections of the bifurcation diagram (cf. the upper left window in Figure 1). A given three dimensional projection of the bifurcation diagram can be viewed from different perspectives using rotations and translations controlled by the mouse or keyboard. Moreover, multiple projections of the same bifurcation diagram can be viewed simultaneously in different windows (not shown). The coordinates plotted in each projection can be interactively selected from the components stored in the data file. The projected curves can also be colored by another component, giving a total of four information fields in each projection.

A final graphics tool allows the user to select a linear combination of two data components as one of the projection coordinates. The linear combination is currently implemented as a rotation in the plane defined by two components of the data file. The tool is especially useful when used to perturb an existing projection by introducing information from a fourth component, in which case it represents a *4D rotation* of the original three dimensional projection.

Each point on the projected bifurcation diagram corresponds to a solution of the BVP. The mouse can be used to explore this correspondence by interactively selecting a point on the bifurcation diagram. The point is then marked, and a projection of the corresponding solution is displayed in another window (cf. the upper right window in Figure 1). All of the visualization tools available in the bifurcation diagram window are also available to manipulate the solution projection.

The data structure in *PCR* could store the full BVP solution. However, if the ODEs are not too stiff, only initial conditions need be stored, since the full solution can be regenerated with an IVP solver. In particular, the reconstruction is fast enough to maintain full graphical interactivity. It is also worth noting that high accuracy reconstruction is generally not needed for visualization purposes. In practice, we numerically solve the BVP and immediately discard all data except for a set of initial conditions that are stored in the *PCR* data file. This is the sense in which *PCR* is a post-processor. The storage savings provided by this regeneration strategy can be substantial. In one example we considered, full storage of the bifurcation diagram required 1,820 MB, but only 1.3 MB were needed for initial conditions. All of the *PCR* examples described here use this 'guaranteed shooting method' for regeneration of the solution to the BVP.

The particular version of *PCR* seen in Figure 1 was implemented using AVS5 (by Advanced Visual Systems Inc.). Another implementation of *PCR* uses the Open Software Foundation's Motif toolkit for the graphical user interface, and the OpenGL graphics system for the three dimensional manipulations. These two libraries were chosen as the basis for the second implementation because of their widespread availability on many workstation architectures.

## 3 Twisted Loops of Elastic Rods

In the next three sections we discuss *PCR* within the context of applications involving elastic rods. A rod model involves a curve in space with an associated frame of directors that describe twist, cf. Figure 2. The equilibrium conditions for a rod can be written as a two-point boundary value problem for a set of seven, second-order (Euler-Lagrange) equations. These equilibrium equations express a balance of the forces and moments acting across each cross-section of the rod. The boundary loadings enter as parameters, and the discretized form of the equations are accordingly of the general type (1).

In this section we consider the problem of a rod that is bent into a loop and then twisted through an angle $\alpha$, cf. Figure 3. For a rod that is isotropic, unshearable, inextensible, and has no inherent curvature or twist, the boundary value problem has explicitly known equilibria involving twisted circles. The full set of equilibrium configurations can then be found using numerical continuation from these known starting points [9]. Part of the resulting bifurcation diagram is depicted in Figure 4. The white ball shown there corresponds to the quadrafoil configuration, shown in Figure 5.[†]

Within the *PCR* data structure, this bifurcation diagram is recorded as arrays of 21-vectors. These vectors contain fourteen initial values for the variables in the BVP, the parameter $\alpha$, and six pre-computed global quantities of interest (e.g., writhe, a singular double integral) that are more conveniently stored than re-computed. *PCR* allows an interactive choice for plotting space curves with any three of the 21 quantities as components, and with a fourth quantity displayed through color, see e.g. Figure 4. Another projection of the same bifurcation diagram can be seen in Figure 6.

Visualization of the bifurcation diagram and corresponding solutions facilitates the understanding of the set of equilibria, and can generate mathematical conjectures. For example, the intricate connectivity of the bifurcation diagram was discovered as a result of visualization. If we denote the number of coverings of the twisted circles as $N$, index the bifurcation points on each straight line by $M$, and denote the straight lines with positive slope as $+$ and the straight lines with negative slope as $-$, then the connectivity of the diagram is given by $(M, N, +) \leftrightarrow (N, M, -)$.

When projecting bifurcation diagrams from higher to lower dimensions, pointwise intersections or multiple coverings of entire branches may appear purely as an artifact of the projection. Such is the case in Figure 4 where all the non-trivial branches are actually double covered. A perturbation using the 4D rotation feature of *PCR* can alleviate this difficulty. For example, Figure 7 is a perturbation of Figure 4 in which the three coordinates of the projection are energy, twist moment, and a linear combination of force and a bending moment. Similarly, Figure 8 shows two branches from Figure 6 with a point of intersection (marked by a white ball), and the same diagram after application of a small 4D rotation.

A projection of the bifurcation diagram where intersections arise only at bifurcation points is described as *topologically correct* [3]. The projection in Figure 7 is topologically correct for

[†]As evinced by the quadrafoil configuration, the formulation of rod equilibria used here does not account for self-contact forces and the rod can pass through itself. More sophisticated codes incorporating self-contact effects have also been developed [8].

example. Topological correctness is one possible criterion for choosing a projection.

Whenever the equilibrium equations are of the gradient form (2), additional information concerning stability, or equivalently the second variation, can be obtained from special *distinguished* projections of the bifurcation diagram [10]. For many problems involving twisted rods, the distinguished diagram is the projection with twist moment plotted vs. angle $\alpha$ [15]. For example, Figure 9 shows the non-trivial $(1, 1, +) \leftrightarrow (1, 1, -)$ branch along with two segments of the single-covered twisted-circle trivial branches. The ratio $\gamma$ of twist stiffness to bending stiffness is crucial to stability exchange in this problem. Figure 9 depicts distinguished diagrams for three values of $\gamma$. The diagram is colored by the constrained index, where red corresponds to index zero (minima) and other colors correspond to non-zero constrained index (saddles). For $\gamma = 1.3$, the entire non-trivial branch is unstable. For $\gamma = 1.7$, the non-trivial branch is stable from the trivial branch to the fold, then unstable until the second fold. For $\gamma = 2.1$, the non-trivial branch is stable. Figure 10 depicts the distinguished diagram with the branch shown in Figure 9 along with several other non-trivial branches. In all cases, the color changes occur at folds, as predicted by the distinguished diagram theory. However, in the projection in Figure 11 which plots twist moment ($M_3$) agains the x-component of force ($F$), the color changes do not occur at folds in the diagram and in fact, do not correspond to any notable feature of the graph. A significant motivation in studying the twisted loop arose from the study of cyclized DNA, as described in the next section.

# 4 Rod model for DNA cyclization

Some DNA molecules are intrinsically curved and have been modelled as elastic rods with inherent curvature [12]. A particularly interesting feature of these curved DNA molecules is their predilection to form cycles [7]. To model this cyclization, we study the same boundary value problem as in Section 3 (cf. Figure 3), but now for rods which are intrinsically curved. We call this an *imperfect problem* as opposed to the *perfect problem* of Section 3.

$PCR$ visualization of the perfect diagram plays a central role in the efficient computation and understanding of the imperfect problem. The perfect diagram (Figure 4) is intricately connected because of the rod's high level of symmetry. For an intrinsically curved rod, these symmetries are broken and the imperfect diagram consists instead of multiple, disconnected components. Since parameter continuation can only trace out solutions on a single component, this disconnectedness poses a challenge in formulating a computational strategy. However, armed with the understanding of the perfect diagram from Section 3, the components of the imperfect problem can be computed efficiently.

We introduce a homotopy parameter $\delta$ which gradually adds intrinsic curvature to the rod, so that $\delta = 0$ corresponds to the perfect rod, and $\delta = 1$ gives the actual DNA curvature. With $PCR$-style visualization, we can understand the splitting of the connected perfect diagram, as shown in Figure 12. The ancestry of the various components of the imperfect diagram is apparent. This then suggests which points on the perfect diagram will generate a particular component of the imperfect diagram after homotopy. Without this understanding, it is difficult to be sure that the component of interest is being computed.

In order to model DNA cyclization experiments, we seek low-energy DNA equilibrium configurations that are also local minima. The energy vs. twist moment projections in Figure 12 make it clear that the lowest-energy configurations lie on the green branch. Furthermore, by plotting the twist moment vs. angle projection of this branch (see Figure 13), we can determine which equilibria are local minima using the test described in Section 3. The green closed loop of solutions (or *isola*) in Figure 12 has a distinguished projection in which $\alpha$ increases from 0 to $4\pi$. The diagram in Figure 13, where $\gamma = 1.1$, contains two folds between which solutions are unstable. If instead $\gamma \gtrsim 1.5$, no folds in angle occur, and we have an isola comprising only stable solutions. In standard stability theory, isolas generally have unstable regions since they must contain folds, but here the multivalued nature of angle allows an entirely stable "loop".

DNA cyclization requires not only that the centerline form a loop, but also that the twist angle, $\alpha$, be an integer multiple of $2\pi$, so that the double helix closes. Thus, only select points on the computed bifurcation diagrams represent cyclized DNA. For example, in Figure 13 the configurations with closed helices are marked and fall in the stable region.

This DNA example illustrates that even when only isolated equilibria are of physical interest, it may be worthwhile to introduce additional parameters (such as the imposed twist angle and the imperfection homotopy parameter) and compute solutions by parameter continuation. Visualization of parameter-continued solutions gives information for organizing solution sets and for determining stability properties which may not be available from algorithms which only compute isolated equilibria.

# 5 Ratcheting and Imperfections

The final example illustrating applications of *PCR* in rod mechanics arose as a computational model of experiments designed to investigate the 'ratcheting' observed when various intravenous surgical procedures (such as angioplasty) are performed using highly elastic guide wires [16]. Ratcheting is a general term used to describe a phenomenon in which there is a highly nonlinear response, including hysteresis and associated discontinuous snaps, between an imposed twist and the elastic response of a highly bent rod. A schematic of the appropriate BVP is provided in Figure 14.

For the purposes of this article, the important point is that this BVP is intrinsically multi-parameter; the distance $r$ between the end points controls the shape of the reference configuration ($\alpha = 0$), and the angle $\alpha$ is varied in any given experimental run. If the rod were exactly transversely isotropic (i.e. had an equal response to bending in any direction) ratcheting would not occur. Instead the rod would respond to the imposed end angle by smoothly rotating everywhere about its own tangent without changing the shape of the centerline. The point of interest is that imperfections involving quite small natural curvatures of the unstressed rod can have dramatic effects on the overall response of the system. Thus, in addition to the two primary parameters $\alpha$ and $r$, the problem has a number of imperfection parameters (in principle an infinite number) describing the intrinsic curvature of the rod.

Figure 15(a) is a one-parameter bifurcation diagram in which the unstressed state was taken to be a shallow circular arc. The angle $\alpha$ was held fixed at zero as the distance between the ends $r$ was varied over both positive and negative values. The diagram is plotted in

$(n_1, n_2, q_3)$ space, with $n_1$ and $n_2$ being two components of force in the rod, and $q_3 = \sin\alpha$. The bifurcation diagram is entirely planar in this projection because $\alpha = 0$ for all of these solutions. Investigation of the diagram using $PCR$ (not illustrated here) reveals that the straight-line branch corresponds to planar configurations, while the curved branch corresponds to non-planar configurations. Each point on this one-parameter bifurcation diagram can be used as the starting point for another one-parameter continuation calculation in which $r$ is held fixed and the angle $\alpha$ is varied. A selection of such curves is depicted in Figure 15(b). Here the branch of part (a) is retained, with the additional branches superimposed.

This example shows that multiple one-parameter computations may not be an efficient way to compute a two dimensional manifold of solutions. While Figure 15(b) gives some sense of the complicated self-intersecting surfaces that can arise in bifurcation diagrams, the computed points are not particularly well distributed over the projected solution surface. In particular it would be better to compute a triangularization of the surface using an inherently two-parameter, adaptive continuation strategy. Similarly, a steered calculation could concentrate computations on the most interesting regions of the bifurcation surface.

# 6 Two-Parameter, Steered Continuation

We now describe a research code by Maddocks and C. K. Mesztenyi that implements two-parameter continuation with a graphical interface providing computational steering. The code is optimized for the interactive exploration of the set of critical points of two-parameter constrained variational principles, and is accordingly called $MC^2$, an acronym for Multiplier and Constraint Continuation. $MC^2$ has been applied to investigate relative equilibria in finite dimensional Hamiltonian systems as arise, for example, in the steady spins of satellites driven by internal momentum wheels (cf. Figure 17). Such systems give rise to constrained variational principles with associated first order conditions of the particular form (3) and $n \approx 10$ unknowns (see [11] for example).

The compute engine in $MC^2$ is a version of a multi-parameter Fortran continuation code of Rheinboldt [13] that was specialized and extended to treat constrained variational problems. The code was tailored to run the numerics interactively from within a three dimensional graphics environment (again based on the AVS5 package). The operator can control the extension directions of the two dimensional manifold of solutions using information extracted from the simultaneous viewing of certain distinguished projections.

An example of the $MC^2$ interface is displayed in Figure 16. The two large windows display user-selectable projections of the two dimensional solution manifold that has been approximated by a numerical calculation based upon a rectangular mesh. The four corners of this mesh are labelled A–D, and the GUI allows extension of any edge of the manifold for a prescribed number of steps. This extension can be done in real time by calling an external computational routine.

The user may also explore the variation of other quantities over the solution manifold through the interactive display of one or more contour plots (the white tubes in Figure 16). In the particular example shown, the coloring of the surface encodes an integer representing the constrained index of each critical point. This index is determined from the signs of the principal curvatures of the solution manifold in certain distinguished projections [11], namely the Hamiltonian projection $(H, C^1, C^2)$ and the Lagrangian projection $(F, \mu_1, \mu_2)$ (cf. the notation introduced in (3)). For example, Figure 16 shows the Lagrangian projection on the left, in which the transition from constrained minima of index 0 (green) to saddles of constrained index 1 (yellow) arises at a line of inflection points in the solution surface. The same transition arises along a cusp line in the Hamiltonian projection shown on the right. In summary, the interactive graphical interface provides immediate access to much of the interesting information concerning the solution set, which can be exploited to steer further numerical exploration of parameter space.

As shown in Figure 17, $MC^2$ can be used to construct quite elaborate bifurcation diagrams. For each component to be computed, the user must supply an initial guess that converges to a solution in that component. The computation can then be steered to compute as much of that component as desired.

Figure 18 shows two distinguished projections of part of the solution manifold for a constrained variational principle governing the steady motions of point vortices. The output illustrates difficulties that arise for multi-parameter continuation that do not arise for one-parameter continuation. The manifold of solutions may have nontrivial topology, for example as created by a singularity in the solution set. Within $MC^2$ it is sometimes then necessary to compute one component of the solution set with multiple meshes generated from multiple initial guesses. At junctions between two meshes, overlap occurs, which is often acceptable, but which is also wasteful of computation. A related issue is that a uniform rectangular mesh in the parameter discretization may not be suitable, perhaps due to the presence of a nearby singularity (cf. projections (a) and (b) in Figure 18). Adaptivity in the parameter step sizes would accordingly be desirable. While some analysis and effective codes addressing these issues for two-parameter problems are available (e.g. [6]), the numerical treatment of nontrivial topology and mesh adaptivity in the computation of multi-parameter solution manifolds still presents challenging computational problems.

# 7 Summary and Future Work

We have surveyed a number of application problems which exploit visualization in their computation. In particular, we have described two software packages, $PCR$ and $MC^2$, developed within our group. $PCR$ focuses on the post-processing of solution sets for one-parameter ODE boundary value problems using a variety of interactive visualization techniques. $MC^2$ focuses on the interactive steering of two-parameter continuation, but is currently only implemented for small problems and does not provide the full assortment of visualization tools found in $PCR$. Accordingly, merging the capabilities of these two codes is an ongoing project. The combined software is called AvA, an acronym for AUTO visualization Application. It is currently implemented for one parameter two-point BVPs and combines the visualization tools of $PCR$ with graphically controlled interactive steering to drive a computational engine based upon a parallel (PVM) implementation of AUTO. AvA runs on local graphics workstation but can control computations on remote computers. Current networking and computing technology make such a project feasible for either one or two-parameter continuation in two-point BVPs.

Two-parameter continuation offers a level of parallelism in addition to the parallelization of the underlying solution algorithm: processors can simultaneously compute solutions for different values of the parameters. In particular, large-scale multi-parameter computations are well-suited for high-performance parallel computing resources, such as parallel supercomputers or workstation farms linked by high-speed networks, e.g. ATM. One of our current objectives is to extend AvA to such two-parameter continuation problems which will represent a complete merging of the capabilities of $MC^2$ and $PCR$.

Extension of this paradigm to more parameters and more spatial variables requires essentially the same numerical routines, but the size of computation grows rapidly. Consequently, interactive remote computation for large problems may also require design of data compression strategies analogous to the one implemented in $PCR$. In addition, visualization of three or higher dimensional manifolds is in itself a topic of continuing research, giving rise to many interesting visualization issues, e.g. appropriate cross-section or projection methods, and exploitation of contour and texture mapping.

**Acknowledgments**: JHM was supported by the AFOSR and ONR; RSM by an NSF Postdoctoral Research Fellowship and the AFOSR; RCP by an AFOSR AASERT Award; and KAR by an ONR AASERT Award. Some computations used facilities provided by the Keck Foundation. It is a pleasure to thank current and former members of LCVM for their contributions to the work surveyed in this article, in particular Yiwei Li, Charles Mesztenyi, and Gabor Domokos.

# References

[1] E. L. Allgower and K. Georg. Continuation and path following. *Acta Numerica*, 22:1–64, 1992.

[2] E. J. Doedel, H. B. Keller, and J. P. Kernévez. Numerical analysis and control of bifurcation problems, parts I and II. *Int. J. of Bif. and Chaos*, 3, 4:493–520, 745–772, 1991.

[3] G. Domokos. Global description of elastic bars. *ZAMM*, 74(4):T289–T291, 1994.

[4] G. Domokos and R. C. Paffenroth. Case study: Visualization for boundary value problems. In R. D. Bergeron and A. E. Kaufman, editors, *IEEE Visualization '94 Conference Proceedings*, pages 345–348. IEEE Computer Society Press, 1994.

[5] G. Domokos and R. C. Paffenroth. *PCR* - a visualization tool for multi-point boundary value problems. Technical Report BN-1167, Institute for Physical Science and Technology, University of Maryland, College Park, 1994.

[6] M. E. Henderson. Computing implicitly defined surfaces: Two parameter continuation. Technical report, IBM Research Division, T. J Watson Research Center, 1993.

[7] J. D. Kahn and D. M. Crothers. Protein-induced bending and DNA cyclization. *Proc. Natl. Acad. Sci. USA*, 1992.

[8] Y. Li and J. H. Maddocks. On the computation of equilibria of elastic rods, Part II: Effects of self-contact. in preparation.

[9] Y. Li and J. H. Maddocks. On the computation of equilibria of elastic rods, Part I: Integrals, symmetry and a Hamiltonian formulation. *J. Comp. Phys.*, to appear.

[10] J. H. Maddocks. Stability and folds. *Arch. for Rat. Mech. and Anal.*, 99(4):301–328, 1987.

[11] J. H. Maddocks and R. L. Sachs. Constrained variational principles and stability in Hamiltonian systems. In Meyer Dumas and Schmidt, editors, *Hamiltonian Dynamical Systems*, volume 63 of *IMA Volume in Mathematics and Its Applications*. Springer Verlag, 1995.

[12] R. S. Manning, J. H. Maddocks, and J. D. Kahn. A continuum rod model of sequence-dependent DNA structure. *J. Chem. Phys.*, 105:5626–5646, 1996.

[13] W. C. Rheinboldt. On the computation of multi-dimensional solution manifolds of parameterized equations. *Numer. Math.*, 53:165–181, 1988.

[14] W. C. Rheinboldt and J. V. Burkardt. A locally-parameterized continuation process. *ACM Trans. Math. Software*, 9:215–235, 1983.

[15] K. A. Rogers. *Stability of Twisted Elastic Rods.* PhD thesis, University of Maryland, College Park, 1997.

[16] J. A. Warner. Numerical simulations in rod mechanics. Master's thesis, University of Maryland, College Park, 1996.

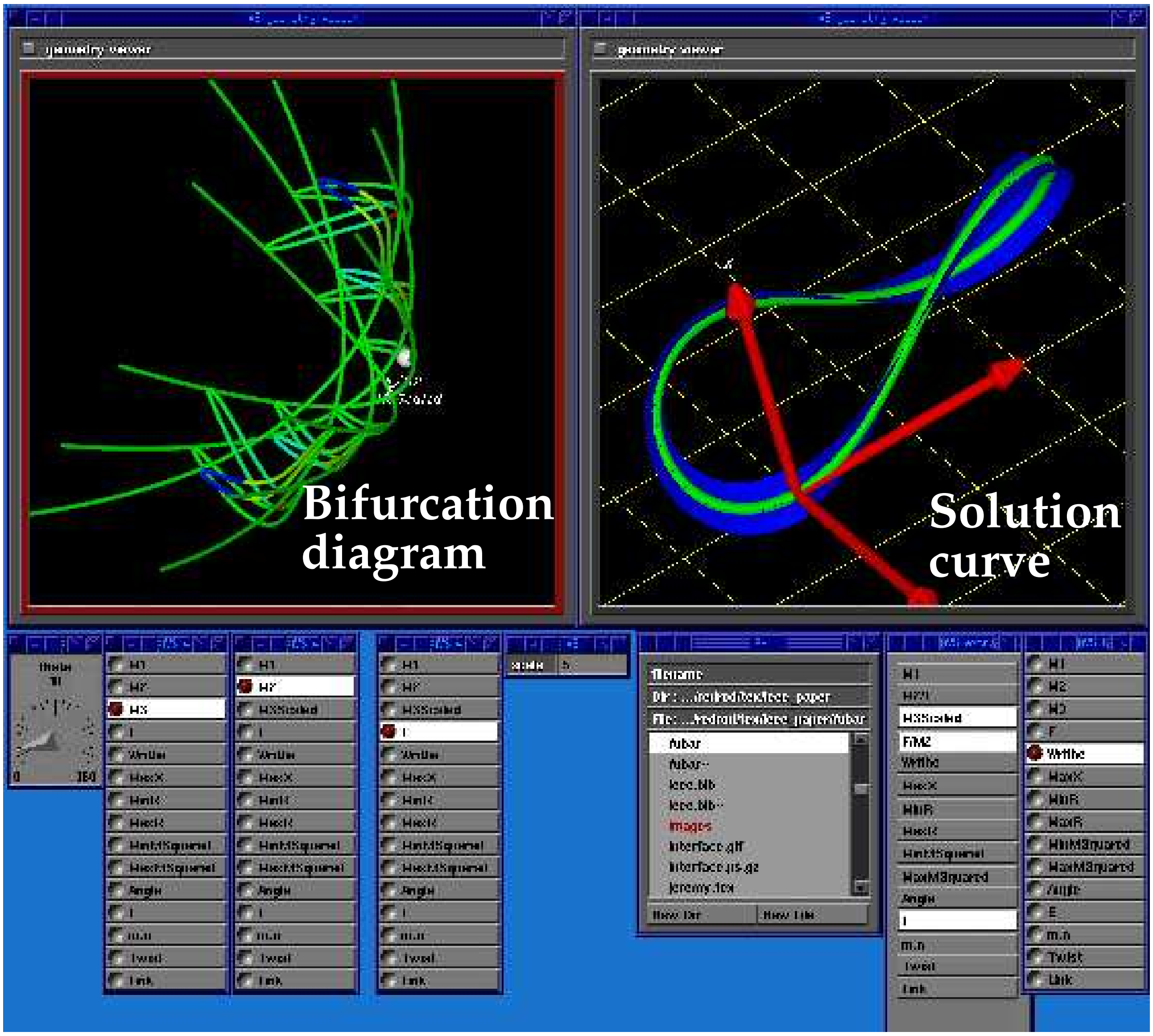


Figure 1: The $PCR$ interface shows the bifurcation diagram projection (on the left) and the solution projection (on the right). The GUI at the bottom allows the user to select both projections.

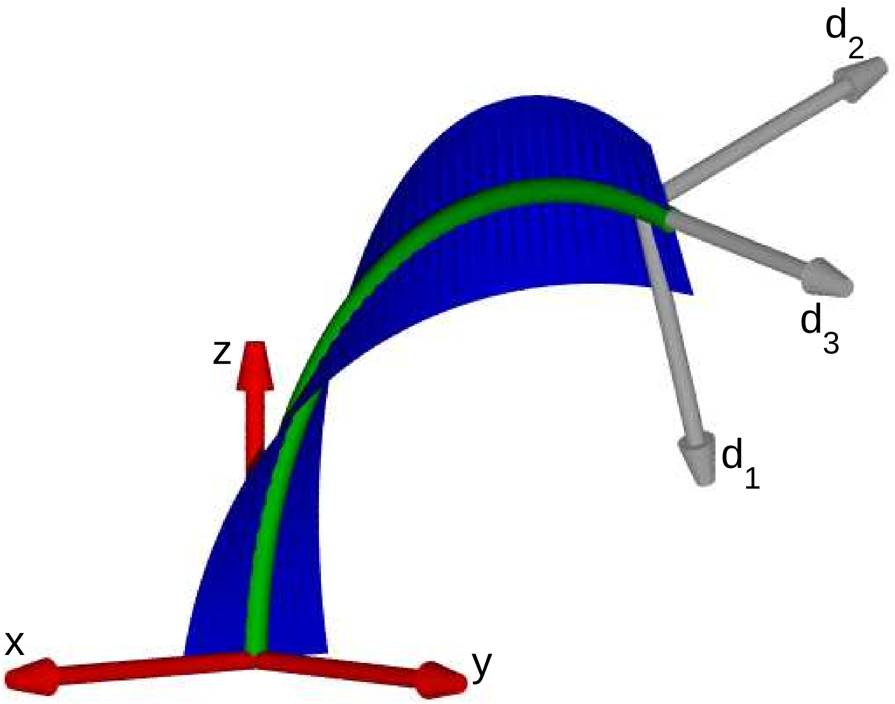


Figure 2: An elastic rod is modelled as a centerline, pictured here as a green tube, along with a frame of directors $(d_1, d_2, d_3)$ that give the orientation of the rod cross-section at each point on the centerline. The director $d_1(s)$ is depicted as a blue ribbon that tracks the twist of the rod.

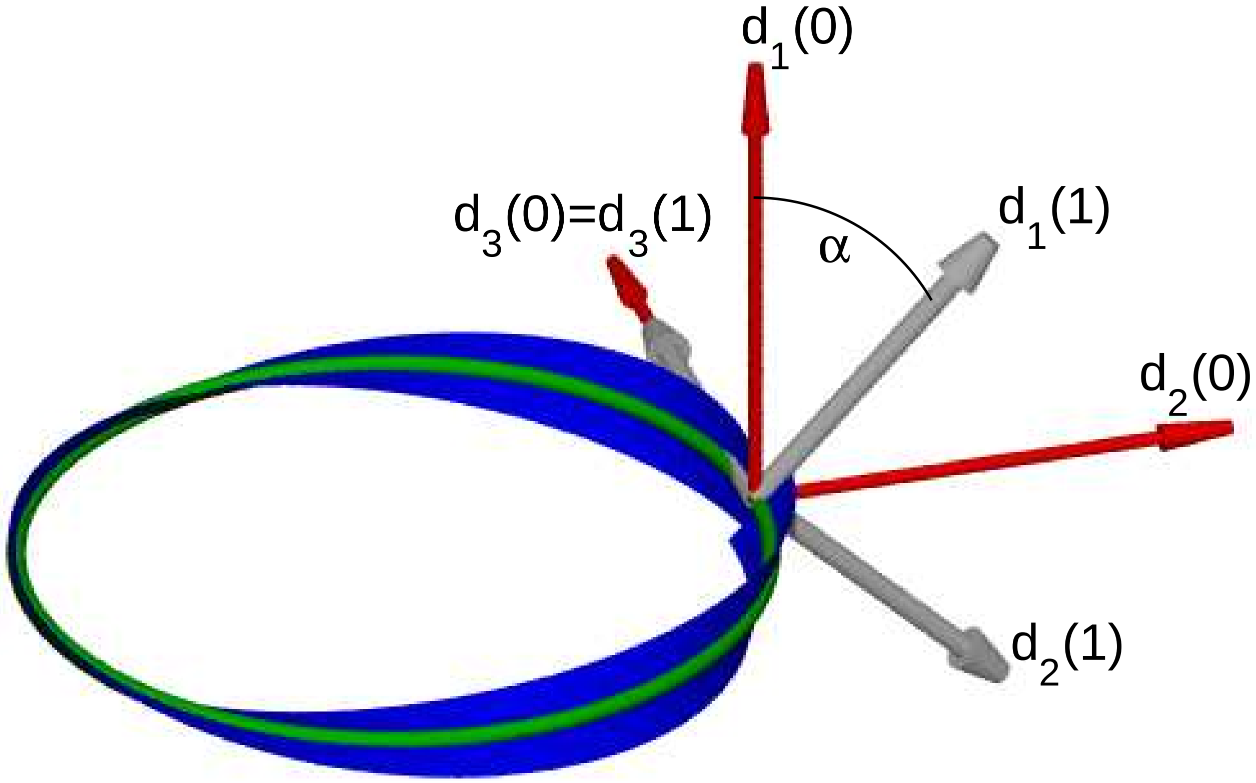


Figure 3: The elastic loop boundary value problem: The directors are shown at the endpoints of the rod; the red axes at $s = 0$ and the gray axes at $s = 1$. The angle $\alpha$ between $d_1(0)$ and $d_1(1)$ is the continuation parameter.

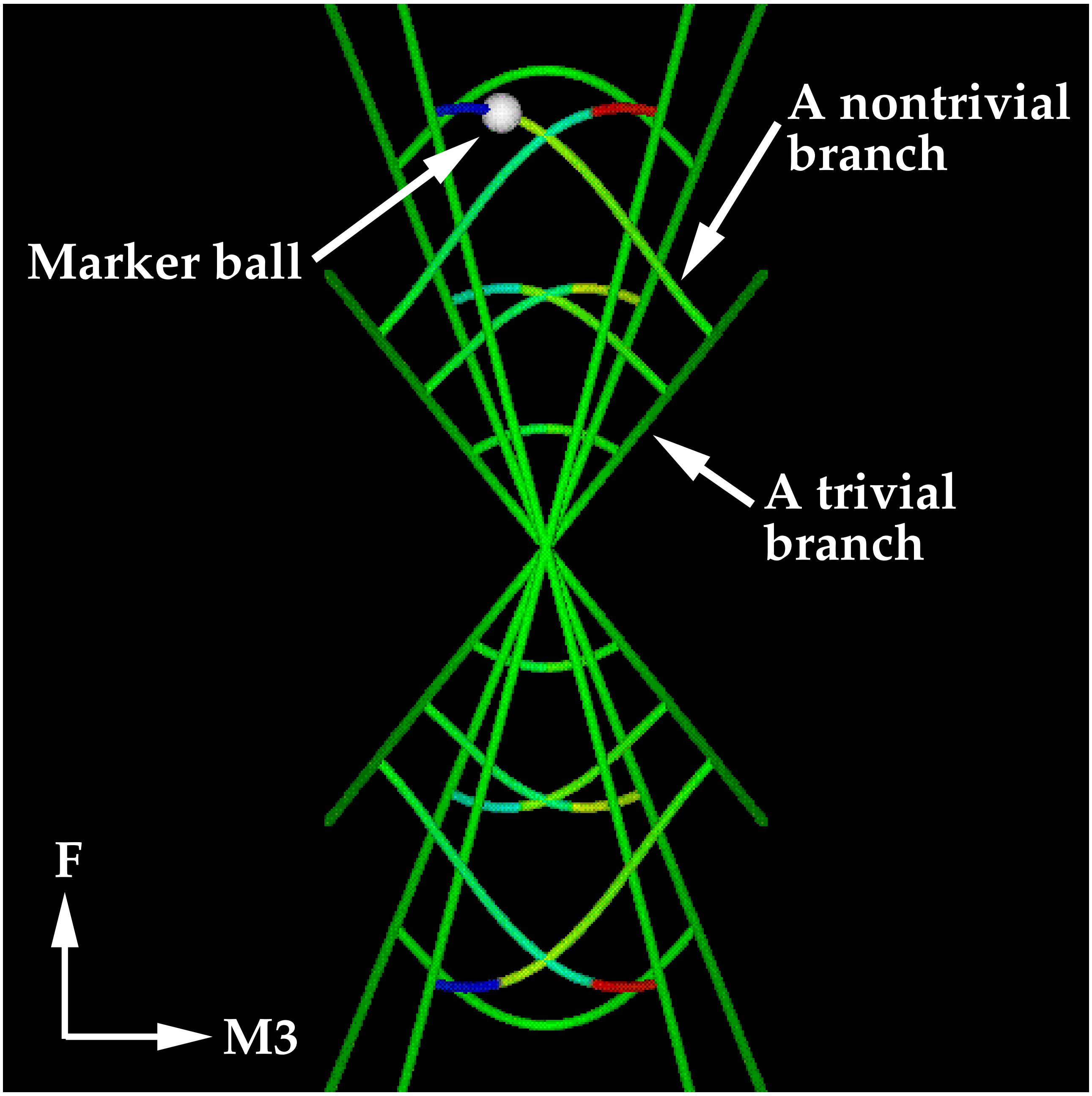


Figure 4: The bifurcation diagram for the elastic loop projected onto the space spanned by the twist moment ($M_3$), $x$-component of force (F), and energy (out of the page): The solution set consists of straight lines connected by curved segments. The straight lines represent trivial branches of planar circular solutions. The curved arcs represent non-trivial branches of non-planar equilibria. The diagram is colored by writhe, which is a measure of non-planarity of the equilibria.

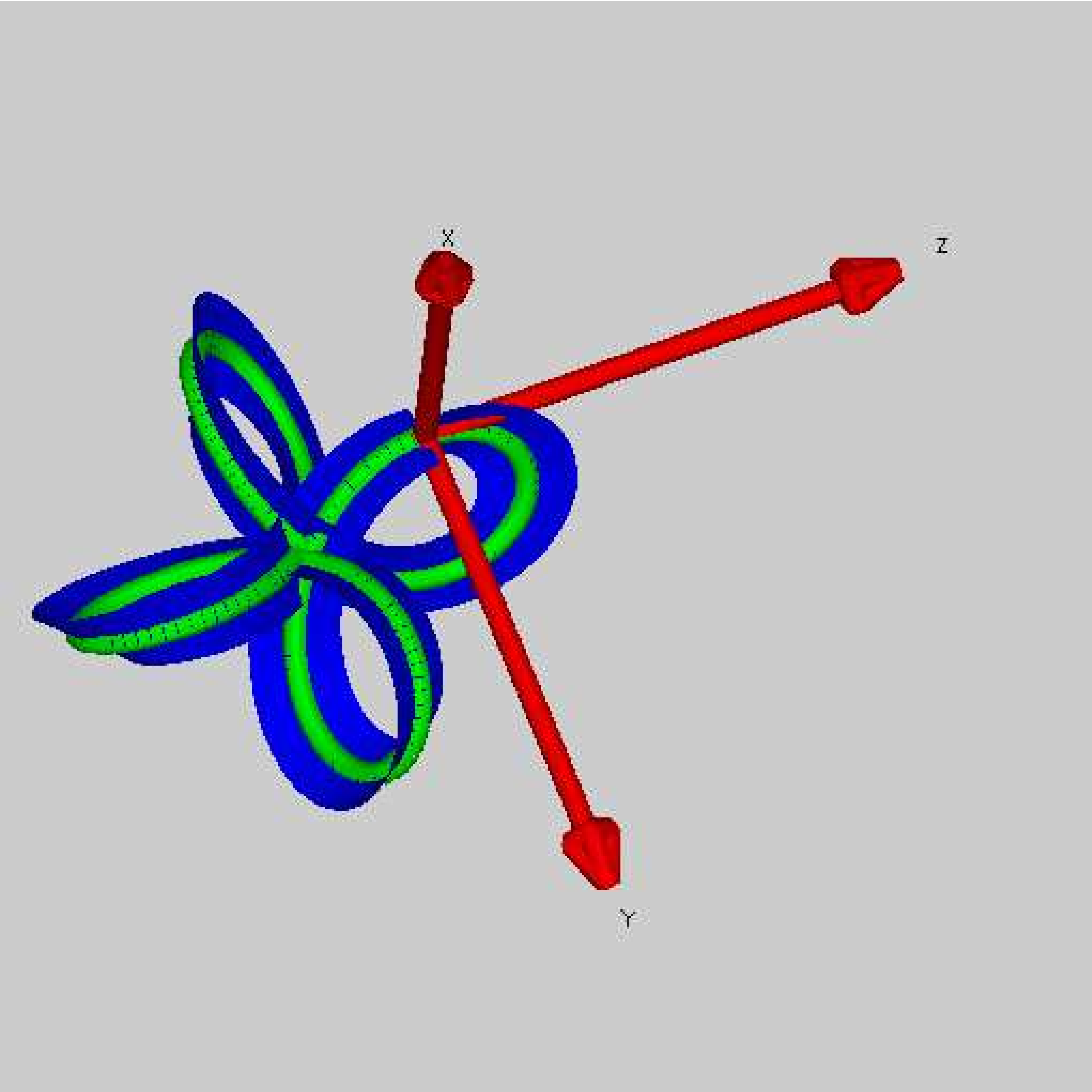


Figure 5: The quadrafoil configuration depicted here corresponds to the white ball in the projected bifurcation diagram shown in Figure 4.

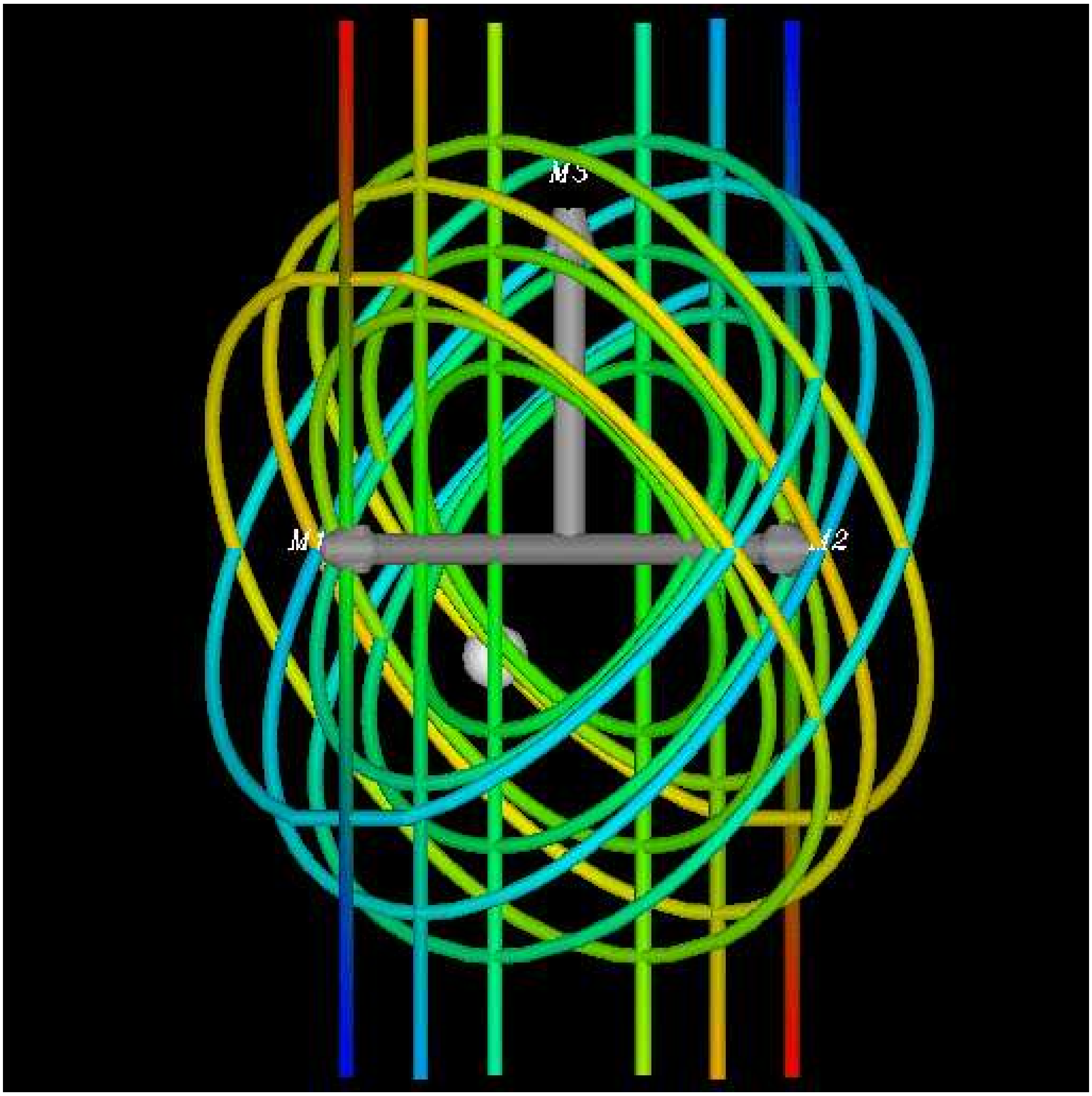


Figure 6: Another projection of the elastic loop bifurcation diagram: The displayed coordinates are two bending moments and the twist moment. The branches are colored by force.

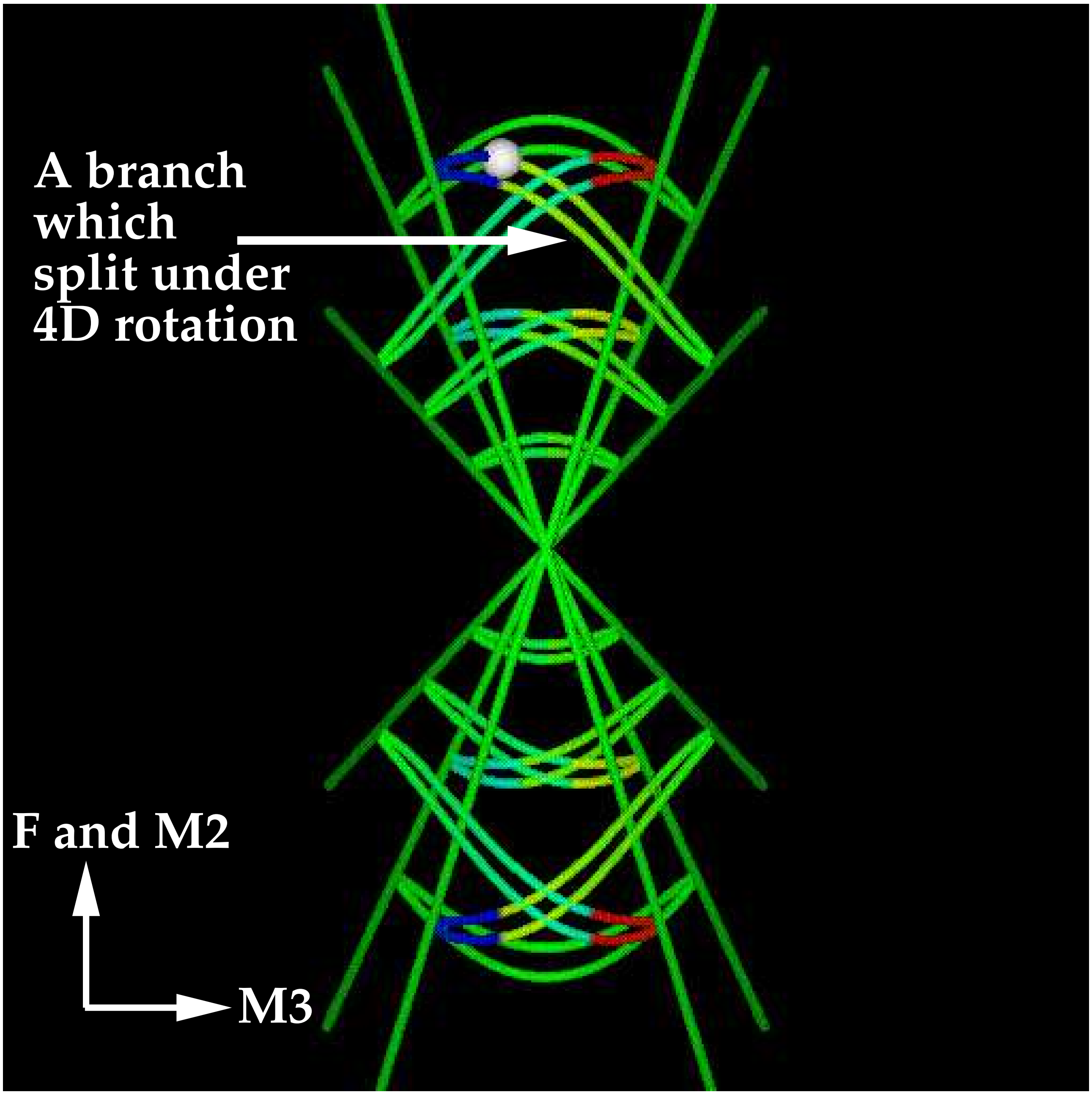


Figure 7: A small 4D rotation breaks the double-covered symmetry of the non-trivial branches in the projection of Figure 4. The resulting diagram is a projection onto the space spanned by energy, twist moment, and a linear combination of the force and a bending moment. A skew view of the same three dimensional projection is shown in Figure 1.

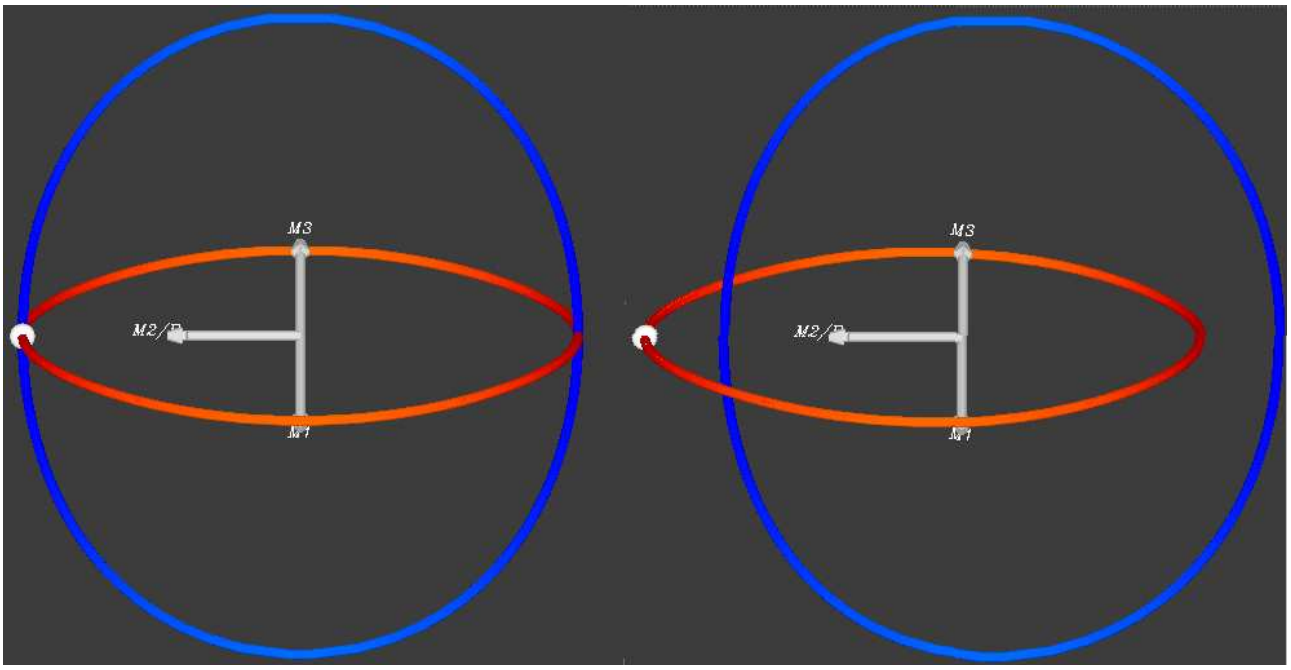


Figure 8: On the left, the white sphere marks a point of intersection that is an artifact of the projection. The point is not a true bifurcation point because it corresponds to two different values of force, as seen by the difference in color of the two branches. On the right, a small 4D rotation separates the two branches.

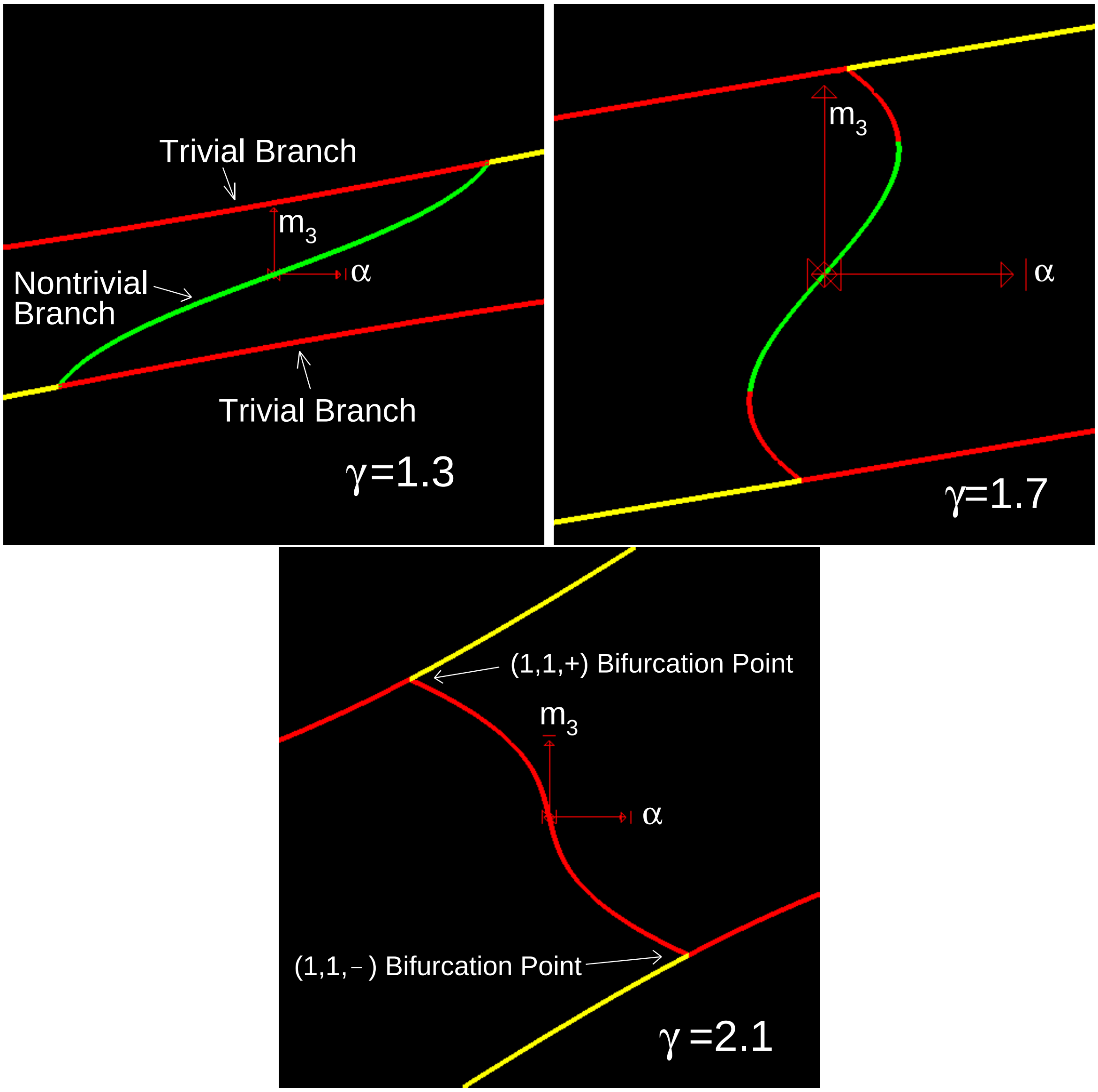


Figure 9: Distinguished diagrams for the elastic loop for varying $\gamma$: The straight lines are the trivial branches. The s-shaped curve is the non-trivial branch connecting the $(1, 1, +)$ to $(1, 1, -)$ bifurcation points. The stability of the non-trivial branch depends on the values of $\gamma$ (see text).

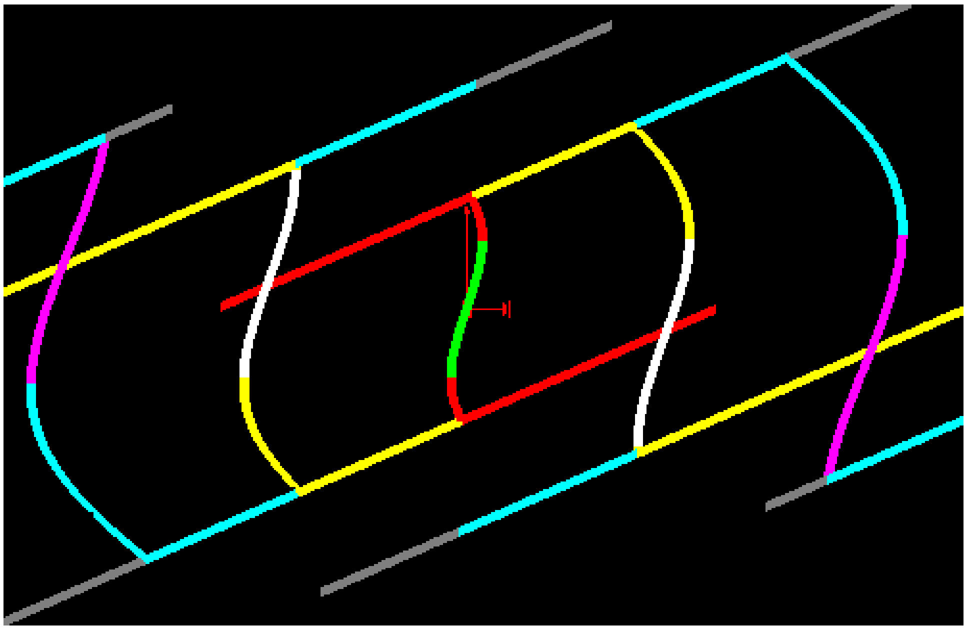

Figure 10: Distinguished diagram for the elastic loop for $\gamma = 1.7$: The straight lines are the trivial branches and correspond to planar equilibria and the s-shaped curves are the non-trivial branches which correspond to nonplanar equilibria. The red segments of the branches represent stable equilibria and other colors represent unstable equilibria.

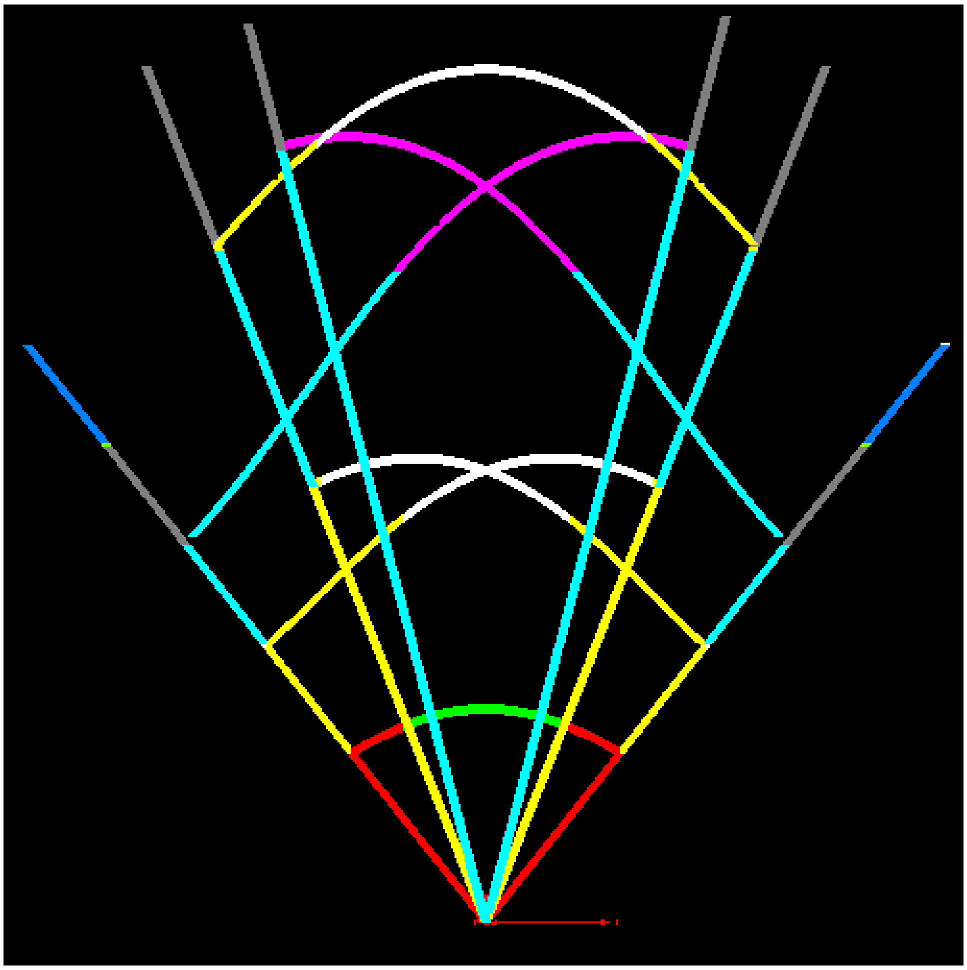

Figure 11: Twist moment ($M_3$) against the x-component of force ($F$) projection of the $\gamma = 1.7$ bifurcation diagram: Colors indicate index as in Figure 10.

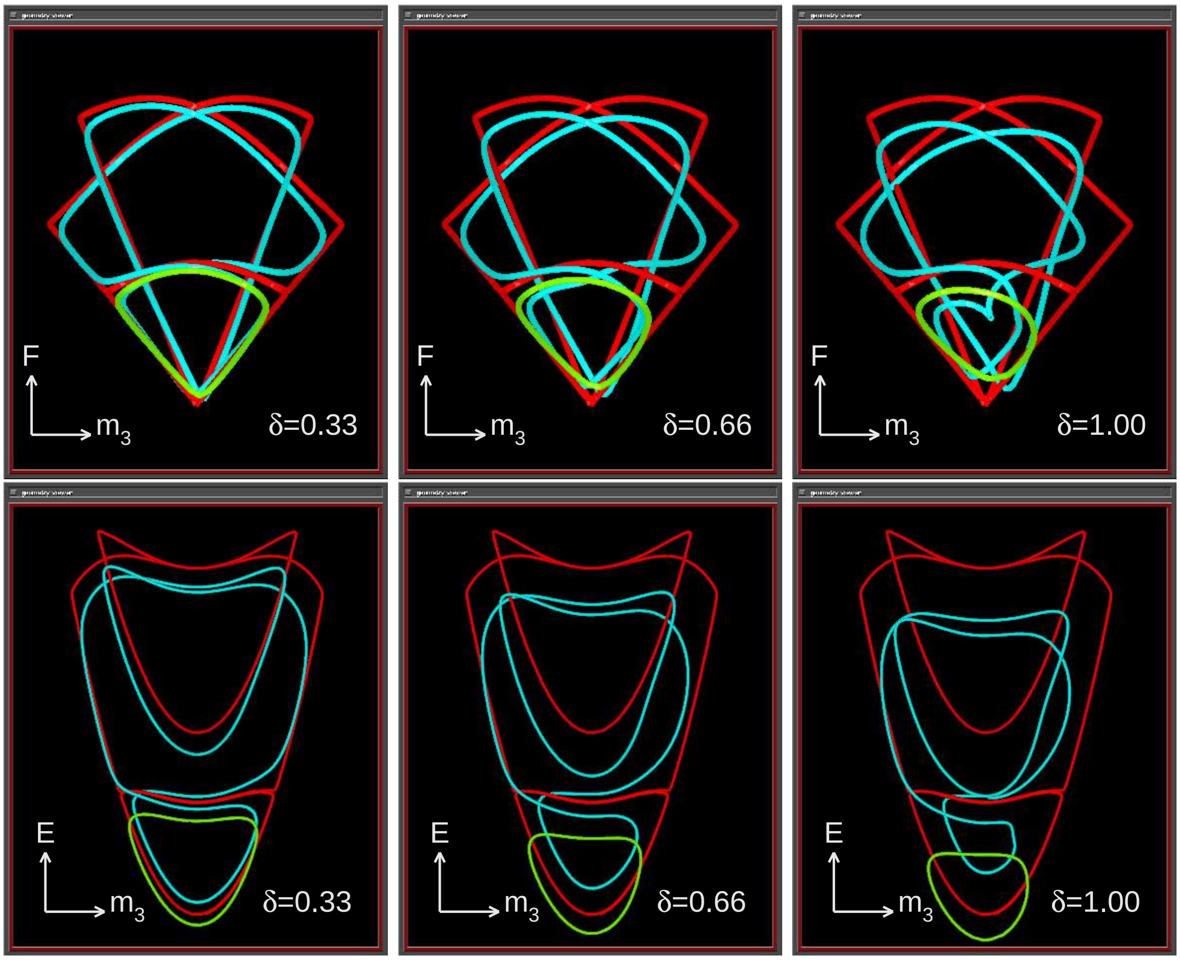


Figure 12: Bifurcation diagrams for intrinsically-curved rods as the homotopy parameter $\delta$ is increased from 0 (no intrinsic curvature) to 1 (full intrinsic curvature). The top series of frames shows the force vs. twist moment projection of the bifurcation diagram (as in Figure 4) and the bottom series shows the energy vs. twist moment projection. The perfect diagram ($\delta = 0$) is shown in all frames in red. Two components of the imperfect diagram are shown in green and blue.

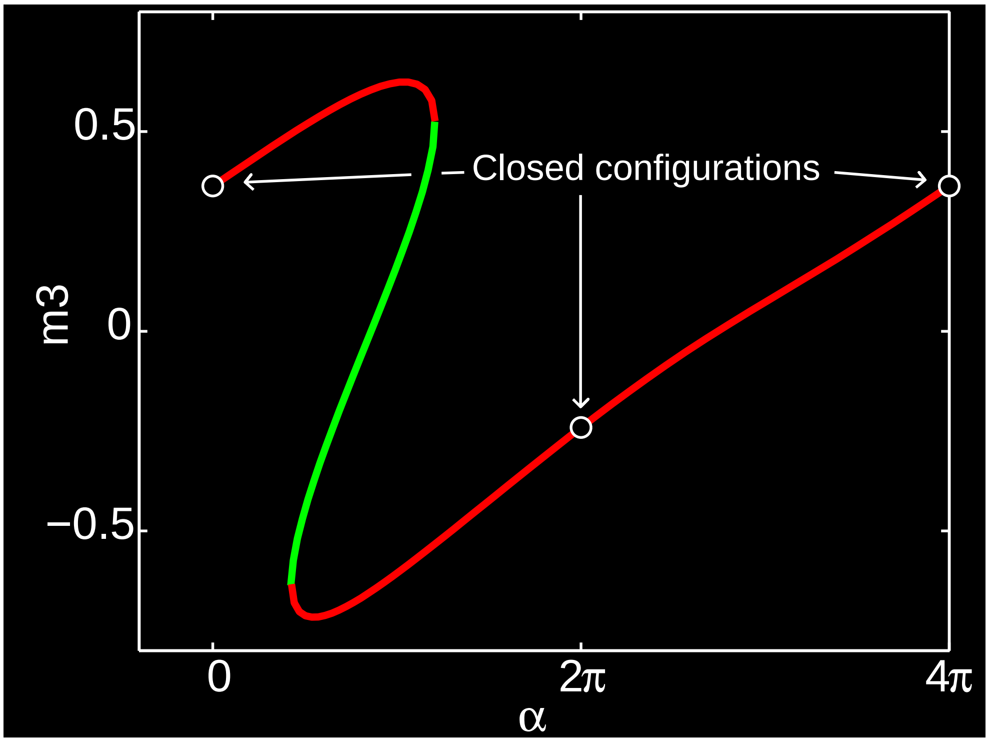


Figure 13: Distinguished diagram for the lowest-energy branch of a DNA molecule with $\gamma = 1.1$: As in the perfect problem, stability changes occur at folds. The figure is qualitatively like the upper right picture in Figure 9, except that the straight lines corresponding to the trivial branches have become slightly curved in this imperfect case. As in Figure 9, there is a region of instability between the two folds. The configurations with closed double helices, marked with circles, fall in the stable region. The closed configurations at $\alpha = 0$ and $\alpha = 4\pi$ are in fact the same solution.

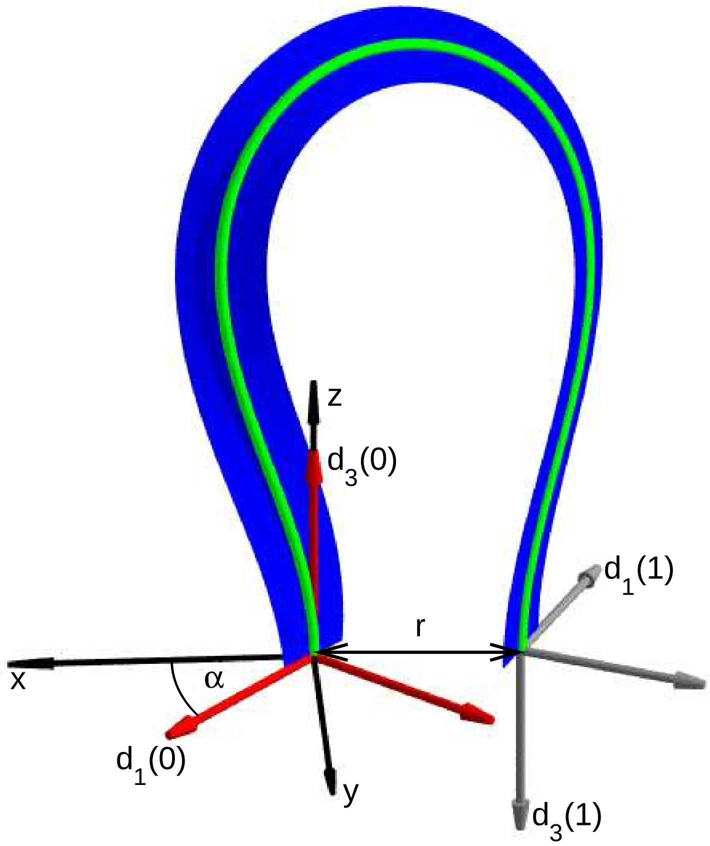


Figure 14: The ratcheting BVP: The ends of a rod are held a distance $r$ from each other, with the end tangents anti-parallel and perpendicular to the line between the ends. A twist angle $\alpha$ is imposed and the elastic response of the rod calculated.

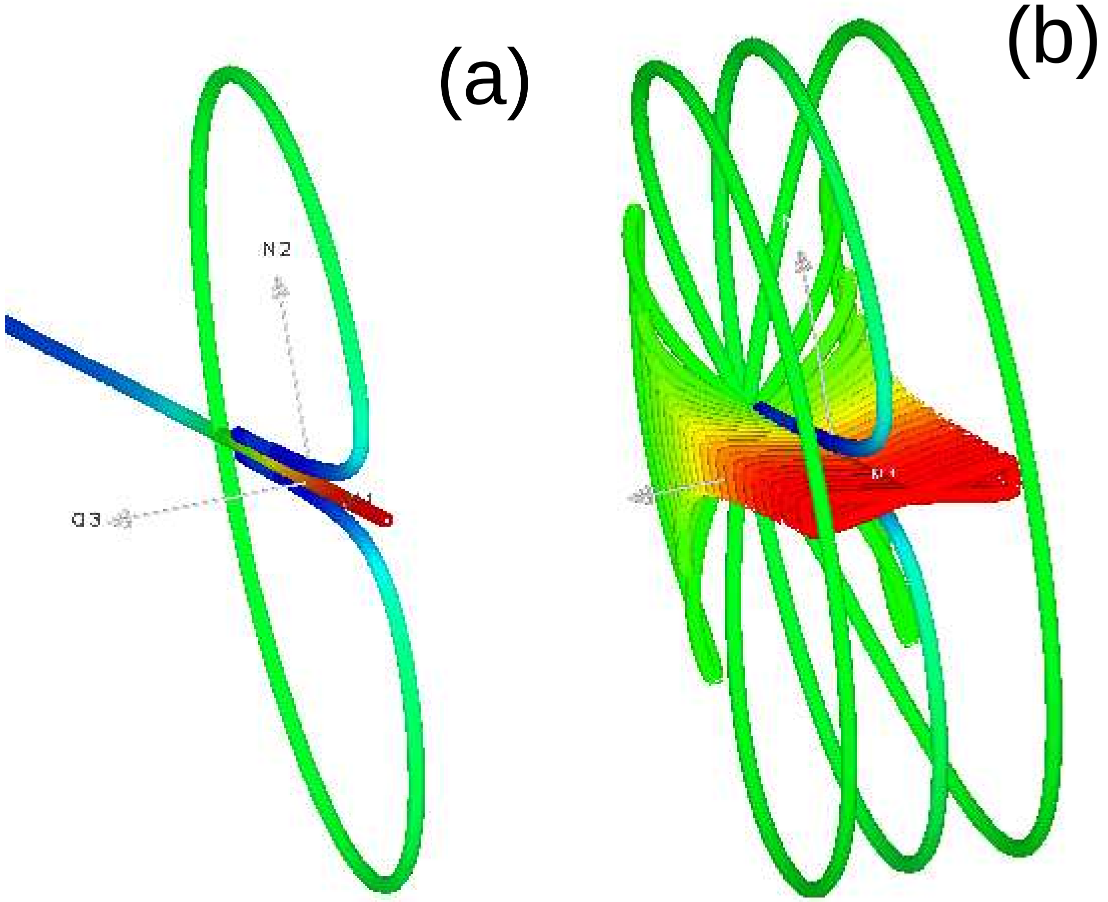


Figure 15: Bifurcation diagrams for the ratcheting BVP: In Figure (a), $\alpha = 0$ is prescribed and $r$ is varied over positive and negative values, to yield a one dimensional manifold of solutions. In Figure (b), twenty additional curves are added by fixing $r$ and varying the angle $\alpha$.

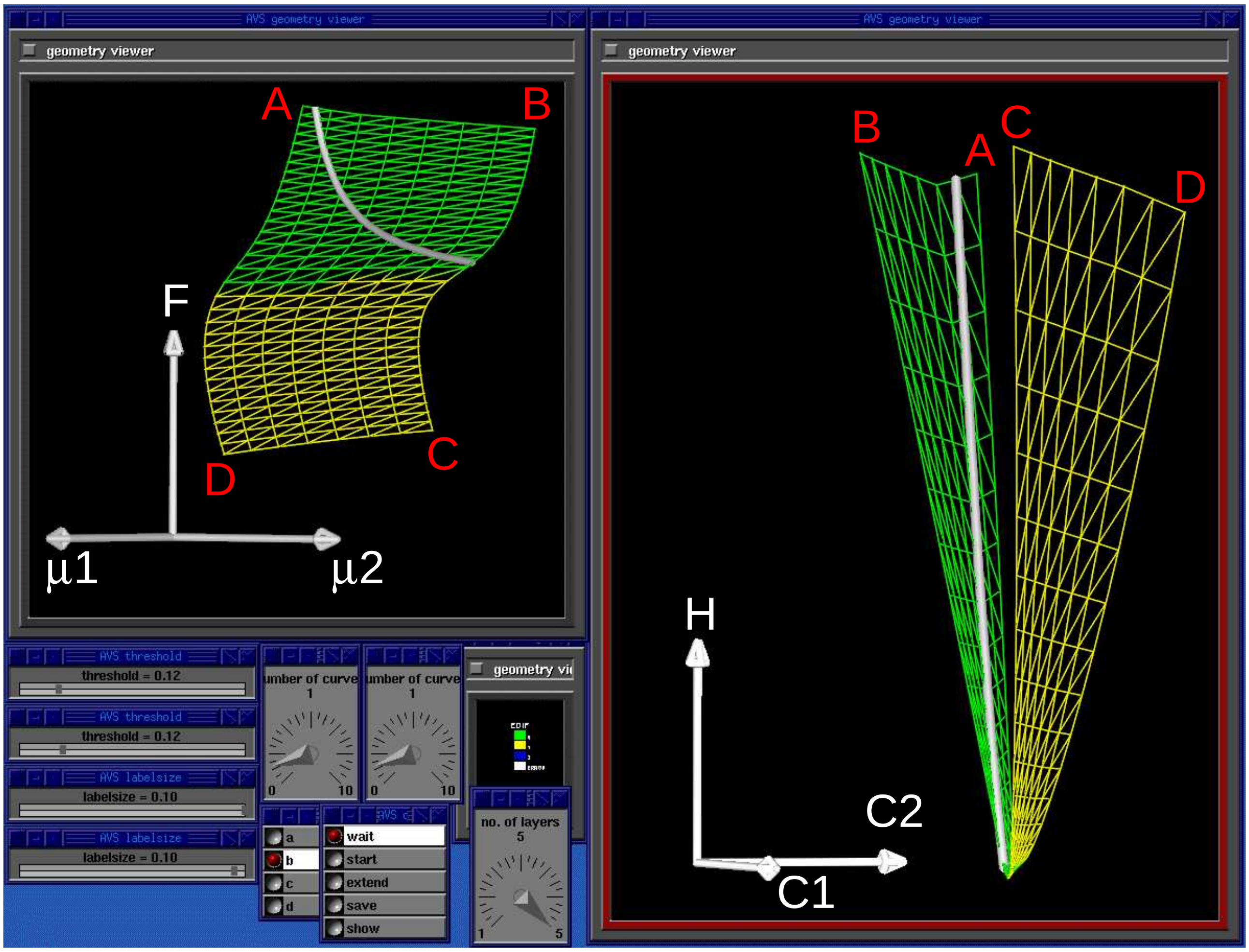


Figure 16: The interface for the steerable, two-parameter continuation code $MC^2$: The two large windows show projections of the solution manifold calculated on a parameter mesh. The smaller panels are part of a GUI used to steer the compute engine and to choose the displayed projections. The computation can be continued by extending the solution surface along any edge.

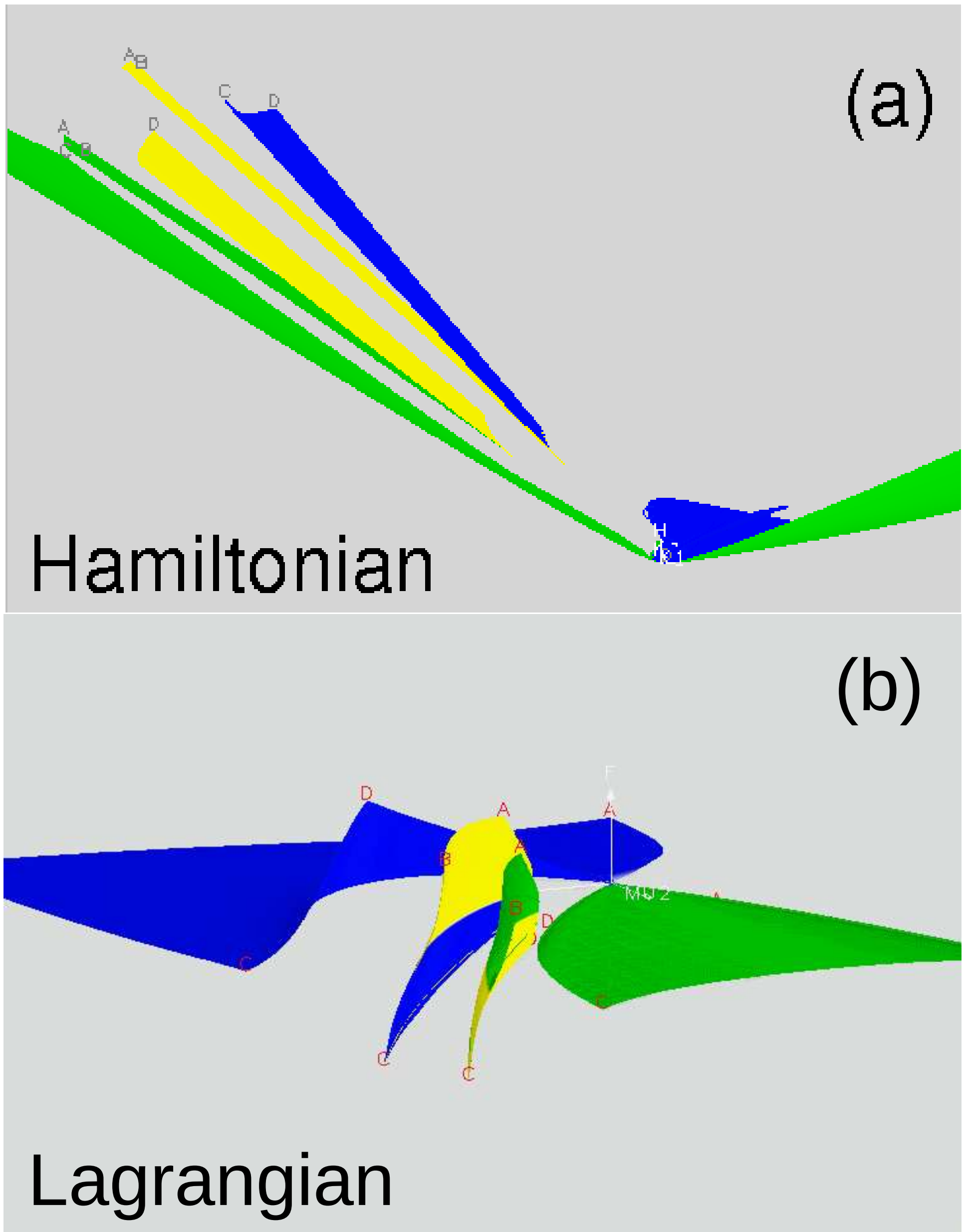


Figure 17: Two distinguished bifurcation diagrams constructed by the $MC^2$ package: In each projection there are four distinct sheets. In projection (a) there is one smooth sheet (entirely green), one self-intersecting sheet (entirely blue), and two bi-color, folded sheets that are cusped. The color coding of the surface is automatically generated by $MC^2$, and indicates the constrained index of each critical point. Notice that the change in constrained index occurs along lines of cusps in projection (a), but along lines of inflection points in projection (b). Some of the features of the projections are comparatively hard to recognize from still views, but are easily visualized when the surfaces are interactively manipulated on the computer screen or seen on video (see for example the MPEG on `http://www.lcvm.umd.edu`.)

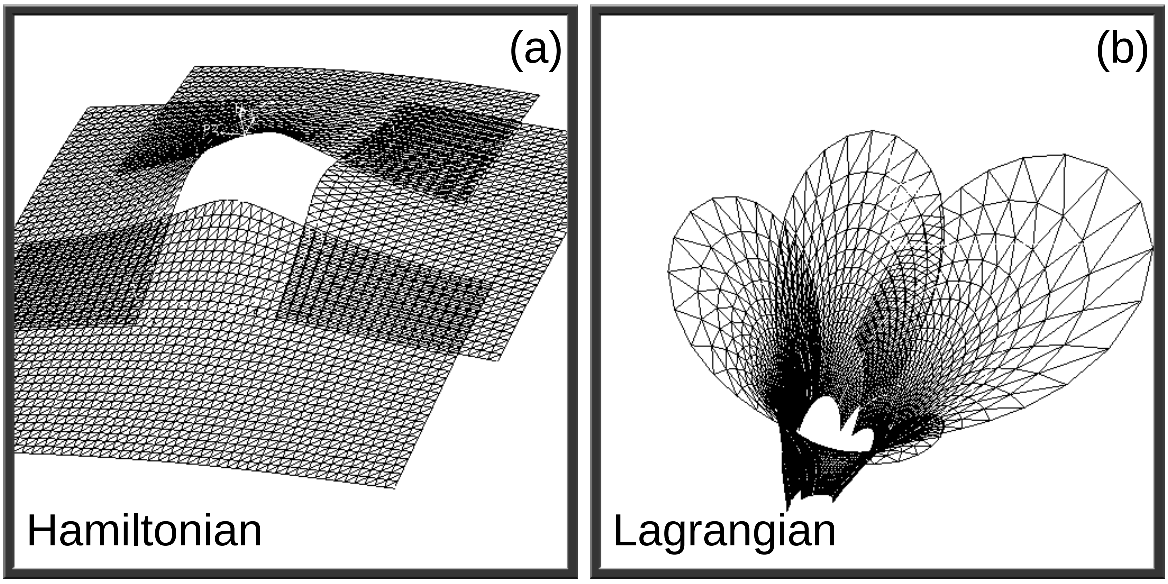


Figure 18: The Lagrangian and Hamiltonian distinguished projections for a point vortex example: The solution manifold contains a singularity, so multiple meshes are required. Additionally, a choice of parameter discretization which yields a regular rectangular grid in the Hamiltonian projection generates a highly distorted grid in the Lagrangian projection. Both of these issues indicate the need for adaptive mesh selection in the multi-parameter continuation routines.