# PRACTICAL NUMERICAL ASPECTS OF MATCONT AND A MODEL FOR CELL CYCLE CONTROL.

Willy Govaerts

Department of Applied Mathematics and Computer Science,

Ghent University,

Belgium.

# Overview

- Matcont, a package for continuation and bifurcation analysis of ODEs.
- Continuation.
- Bifurcations.
- Branch switching.
- Limit cycles.
- Use of archived data.
- Budding yeast example.
- Budding yeast: discussion.

MatCont is a continuation-based matlab toolbox for local and global bifurcation analysis of ODEs.
Developed by WG and Yu. A. Kuznetsov (Utrecht, NL)
Freely available at

**www. sourceforge.net**

as part of the project

**matcont**

(includes bifurcation studies of discrete dynamical systems).

**http://sourceforge.net/projects/matcont**

History starts with A. Riet (2000) and W. Mestrom (2002): master theses in Utrecht.

**Developers:**

1. Annick Dhooge.
2. Bart Sautois.
3. Hil G.E. Meijer.
4. Reza Khoshsiar Ghaziani.
5. Virginie De Witte.

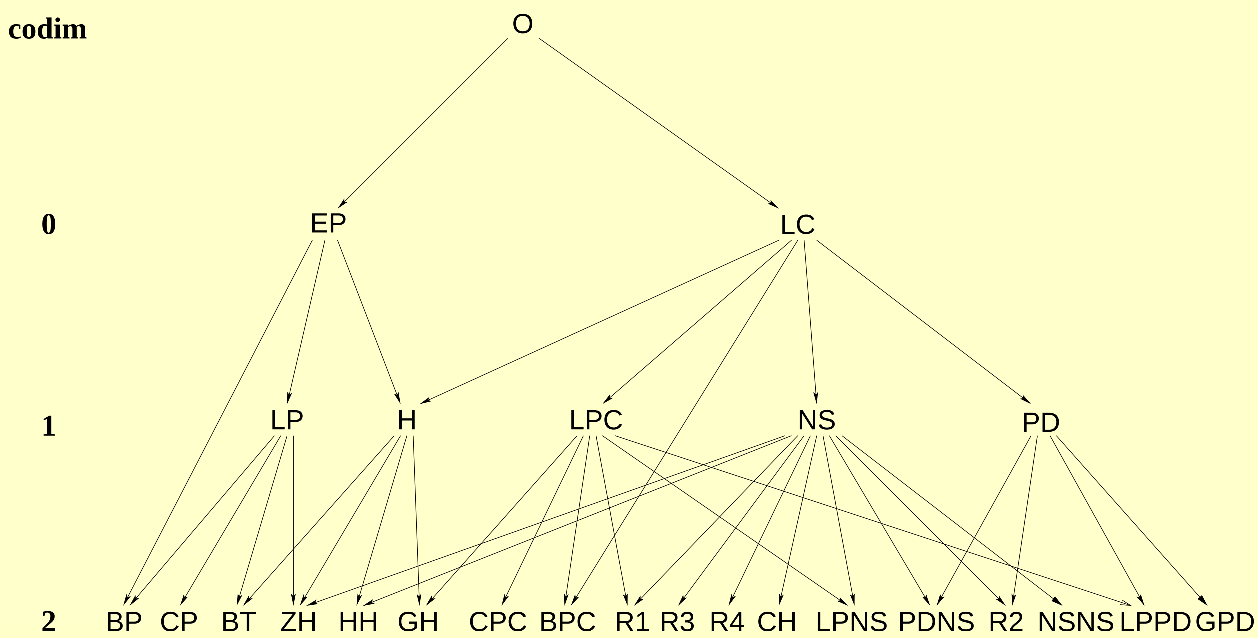
Advice and help of many others, e.g. E. Doedel, V. Govorukhin, A. Steindl.

**Download statistics.**

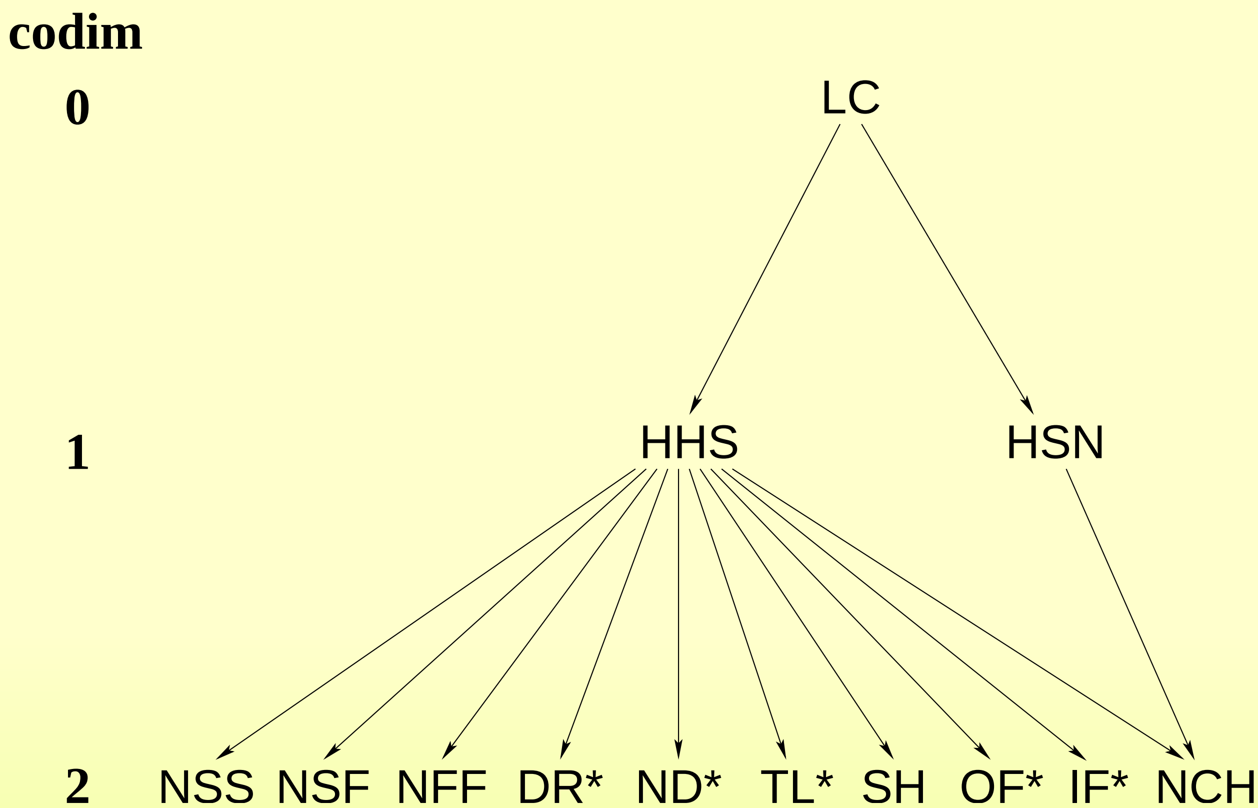
Period: January 1 to March 29, 2009.

1. Cl_MatCont: "Command Line Continuation package in Matlab", 5.5 downloads per day.
2. MatCont: "Continuation in Matlab", same package with GUI, 11.7 downloads per day.
3. Cl_MatContM: "Command Line Continuation package in Matlab for maps", related package for bifurcations of maps, 3.2 downloads per day.

Bifurcations of equilibria and periodic orbits.



Bifurcations of homoclinic orbits.



Continuation keeps track of

1. A point on a curve.
2. A tangent vector along the curve (implies a direction).
3. A stepsize.

Continuation consists of

1. Prediction along tangent.
2. Pseudo-Newton correction to the curve, initially in the hyperplane orthogonal to the tangent.
3. Adaptation of stepsize if appropriate.
4. Storage of new point, tangent vector and stepsize.

Warning: if in the first point no tangent is given, then successive unit vectors are tried as kandidate directions and rejected if no convergence is obtained.

Problem 1: If the number of continuation variables is large, then initialization can take a long time and still fail.

Note: not uncommon.

Problem 2: Initial accepted point on the curve is not close to the given first point.

Note: not uncommon.

Problem 3: Convergence to a faraway point, perhaps on a different branch. Typically happens if the curve is orthogonal to the first unit vector.

Note: problem is rare.

Example: Toy model 1 for the cell cycle.

Note: Can be remedied by first computing an optimal guess for the tangent direction.

Continuation of solutions to $F(X) = 0$, $\dim(X) = \dim(F(X)) + 1$ :

1. $\|F(X)\| \le FunTolerance$,
2. $\|\delta X\| \le VarTolerance$.

General rules:

1. It is more important to have a small *FunTolerance* than a small *VarTolerance*, since $\|\delta X\|$ depends on the conditioning of the problem.
2. Norms are computed in the continuation spaces, not in phase space, so *FunTolerance* and *VarTolerance* as a rule should be larger for limit cycles, homoclinic orbits etc than for equilibria, limit points etcetera.

Detection of bifurcations: based on change of sign of testfunctions $T(X)$.
Location of bifurcations by default uses bisection with convergence criteria:

1. $\|T(X)\| \leq TestTolerance$,
2. $\|X_2 - X_1\| \leq 100 VarTolerance$.

where $T(X)$ changes sign in $]X_1, X_2[$.
Pitfall: there is a certain contradiction in the use of VarTolerance both in computing points on the curve and in deciding convergence of the iteration for a zero of the testfunction.
Factor 100 is arbitrary and hardcoded in line 934 in **Continuer/cont.m.**

For branch points of equilibria the branch point is not a regular zero of the test function.
Specific locators are provided for branch points of equilibria and limit cycles, based on extended systems for which the branch point is a regular zero.
Pitfall: they require that the initial approximation is good enough and the linear system is sufficiently well-conditioned.
Use of specific locators is hardcoded in

- Line 81 of **Equilibrium/equilibrium.m** (for equilibria).
- Line 111 of **LimitCycle/limitcycle.m** (for limit cycles).

(replace 1 by 0 to disable the use of specific locators).

Example (Yu. A. Kuznetsov):

$$\begin{cases} x' &= x^2 + y^2 + \alpha \\ y' &= 2xy, \end{cases}$$

or

$$X' = f(X, \alpha)$$

where

$$X = \begin{bmatrix} x \\ y \end{bmatrix}.$$

Equilibrium curve starts from ($x = 1, y = 0, \alpha = -1$).

Branch point at $(0,0,0)$ is not found by the specific branch point locator if symbolic derivatives are used in the computation of the Jacobian and Hessian. It is sometimes located if finite differences are used. It is always located if the specific locator for branch points is disabled.
Defining system for branch points is

$$\begin{cases} f(X,\alpha) + bp & = & 0 \\ f_{(X,\alpha)}(X,\alpha)^T p & = & 0 \\ p^T p - 1 & = & 0. \end{cases}$$

where

- $p = \begin{bmatrix} p_1 \\ p_2 \end{bmatrix}$ is the left singular vector of $f_{X,\alpha}(X, \alpha)$ at the branch point.
- $b \in \mathbb{R}$ is an auxiliary variable that is initially zero and again zero after convergence.

$$f_{X,\alpha}(X, \alpha) = \begin{bmatrix} 2x & -2y & 1 \\ 2y & 2x & 0 \end{bmatrix}$$

Jacobian of the system:

$$\begin{bmatrix} 2x & -2y & 1 & p_1 & b & 0 \\ 2y & 2x & 0 & p_2 & 0 & b \\ 2p_1 & 2p_2 & 0 & 0 & 2x & 2y \\ 2p_2 & -2p_1 & 0 & 0 & -2y & 2x \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 2p_1 & 2p_2 \end{bmatrix}$$

Non-singular if and only if $p_2 \neq 0$!

$$f_{X,\alpha}(X,\alpha) = \begin{bmatrix} 2x & -2y & 1 \\ 2y & 2x & 0 \end{bmatrix}$$

Exact left singular vector is

$$\begin{bmatrix} 0 \\ 1 \end{bmatrix},$$

Initial choice of $p$ : Eigenvector corresponding to the in absolute value smallest eigenvalue of the state variable block in

$$\begin{bmatrix} 2x & -2y & 1 \\ 2y & 2x & 0 \end{bmatrix}$$

In this case, there are (nongenerically) two equal smallest eigenvalues and Matlab picks the eigenvector

$$\begin{bmatrix} 1 \\ 0 \end{bmatrix},$$

the only bad choice that is possible!

Note: problem can probably be remedied by using the left singular vector that corresponds to the smallest singular value of

$$\begin{bmatrix} 2x & -2y & 1 \\ 2y & 2x & 0 \end{bmatrix}$$

instead of an eigenvector.

Branch switching can fail if the branch point is not located to sufficient accuracy or if the amplitude is not appropriate.

1. To compute the branch point to higher accuracy is always a good idea.
2. Increasing of decreasing the amplitude can both be useful.

Starting a limit cycle: either

1. From a Hopf point.
2. From a stable periodic orbit found by time integration, using the Select Cycle button in the Starter window of MatCont.

A Hopf point is a branch point for the system that defines periodic orbits.
A stable periodic orbit is usually a regular solution to the same system.
Advice 1: the second approach is usually more robust.

Example: it is hard to start from the third Hopf point in the Budding Yeast model (-3) but easy to start from the orbit computed for $m = 1$.

Note: it is sometimes possible to start **unstable** periodic orbits in this way in planar system or systems with only two fast variables.
Advice 2: for a first run switch detection of cycle bifurcations off and monitor multipliers instead.

Limit cycles and their test functions tend to lose accuracy when approaching a homoclinic orbit. This leads to many false detections of LPC, PD, NS and BPC.
Advice:

- Switch off the detections or at least monitor multipliers.
- Repeat computations for several numbers of test intervals.
- If the period is very large, select the periodic orbit and declare it to be a homoclinic orbit (to saddle or to saddle-node).

MatCont is an open system: the outcome of each run is archived as a Matlab **.mat** file. Such a file can be loaded to provide Matlab data for other applications.

Two types of computational tasks:

- Orbit computation run (time is free).
- Continuation run (one or more parameters are free).

Orbit run: **load P_O(2)** loads variables **ctype, option, param, point, s, t, x** where

- **t** is a vector of timepoints.
- **x** is an array whose i-th column is the i-th computed point along the orbit (at time **t**(i).)
- **param** contains the values of the system parameters (fixed in an orbit computation.)

Continuation run, e.g. in the case of limit cycles: **load LC_LC(1)** loads variables **cds, ctype, f, h, lds, num, point, s, v, x** where

- **x** is an array whose i-th column is the i-th computed point along the curve.
- **v** is an array whose i-th column is the normalized tangent vector to the curve.
- **s** is a cell array whose i-th entry is a cell that contains information relevant to the i-th computed special point (like imaginary part of a Hopf eigenvalue, relevant normal form coefficients)

- **h** is an array whose i-th column contains some information on the i-th computed point: stepsize taken in the initial approximation, half the number of Newton iterations (rounded up), values of test functions for bifurcations.
- **f** is an array whose i-th column is related to the i-th computed point but also depends strongly on the type of curve. E.g. for limit cycles it contains the time mesh and the multipliers.
- **cds, lds** are structures that contain details on the computation of the curve. **lds.P0** contains the state variables and parameters in the first point of the orbit.

Cell cycle consists of

$$G1 \to S \to G2 \to M \to G1 \ldots$$

phases.

J.J. Tyson, B. Novák, A. Csikasz-Nagy and many collaborators: apply bifurcation theory and numerical bifurcation studies to explain behaviour under parameter variation.

Basic idea: $G1$ and $S - G2 - M$ correspond to equilibria of the cell system through which the cell is driven by alternation of two antagonistic substances, namely cyclins and APC's (anaphase promoting complexes).

**Budding yeast equations for a fixed cell mass $m$**
**(Tyson and Novák 2002)**

$$\frac{d[CycB]_T}{dt} = k_1 - (k_2' + k_2''[Cdh1] + k_2'''[Cdc20]_A)[CycB]_T,$$

$$\frac{d[Cdh1]}{dt} = \frac{(k_3' + k_3''[Cdc20]_A)(1-[Cdh1])}{J_3 + 1 - [Cdh1]} - \frac{(k_4 m[CycB] + k_4'[SK])[Cdh1]}{J_4 + [Cdh1]},$$

$$\frac{d[Cdc20]_T}{dt} = k_5' + k_5''\frac{(m[CycB])^n}{J_5^n + (m[CycB])^n} - k_6[Cdc20]_T,$$

$$\frac{d[Cdc20]_A}{dt} = \frac{k_7[IEP]([Cdc20]_T - [Cdc20]_A)}{J_7 + [Cdc20]_T - [Cdc20]_A} - \frac{k_8[Mad][Cdc20]_A}{J_8 + [Cdc20]_A} - k_6[Cdc20]_A,$$

$$\frac{d[IEP]}{dt} = k_9 m[CycB](1 - [IEP]) - k_{10}[IEP]),$$

$$\frac{d[CKI]_T}{dt} = k_{11} - (k'12 + k''_{12}[SK] + k'''_{12}[CycB])[CKI]_T,$$

$$\frac{d[SK]}{dt} = k'_{13} + k''_{13}[TF] - k_{14}[SK],$$

$$\frac{d[TF]}{dt} = \frac{(k'_{15}m + k''_{15}[SK])(1 - [TF])}{J_{15} + 1 - [TF]} - \frac{(k'_{16} + k''_{16}m[CycB])[TF]}{J_{16} + [TF]},$$

where

$$[CycB] = [CycB]_T - [Trimer],$$

$$[Trimer] = \frac{2[CycB]_T[CKI]_T}{\Sigma + \sqrt{\Sigma^2 - 4[CycB]_T[CKI]_T}},$$

$$\Sigma = K_{eq}^{-1} + [CycB]_T + [CKI]_T.$$

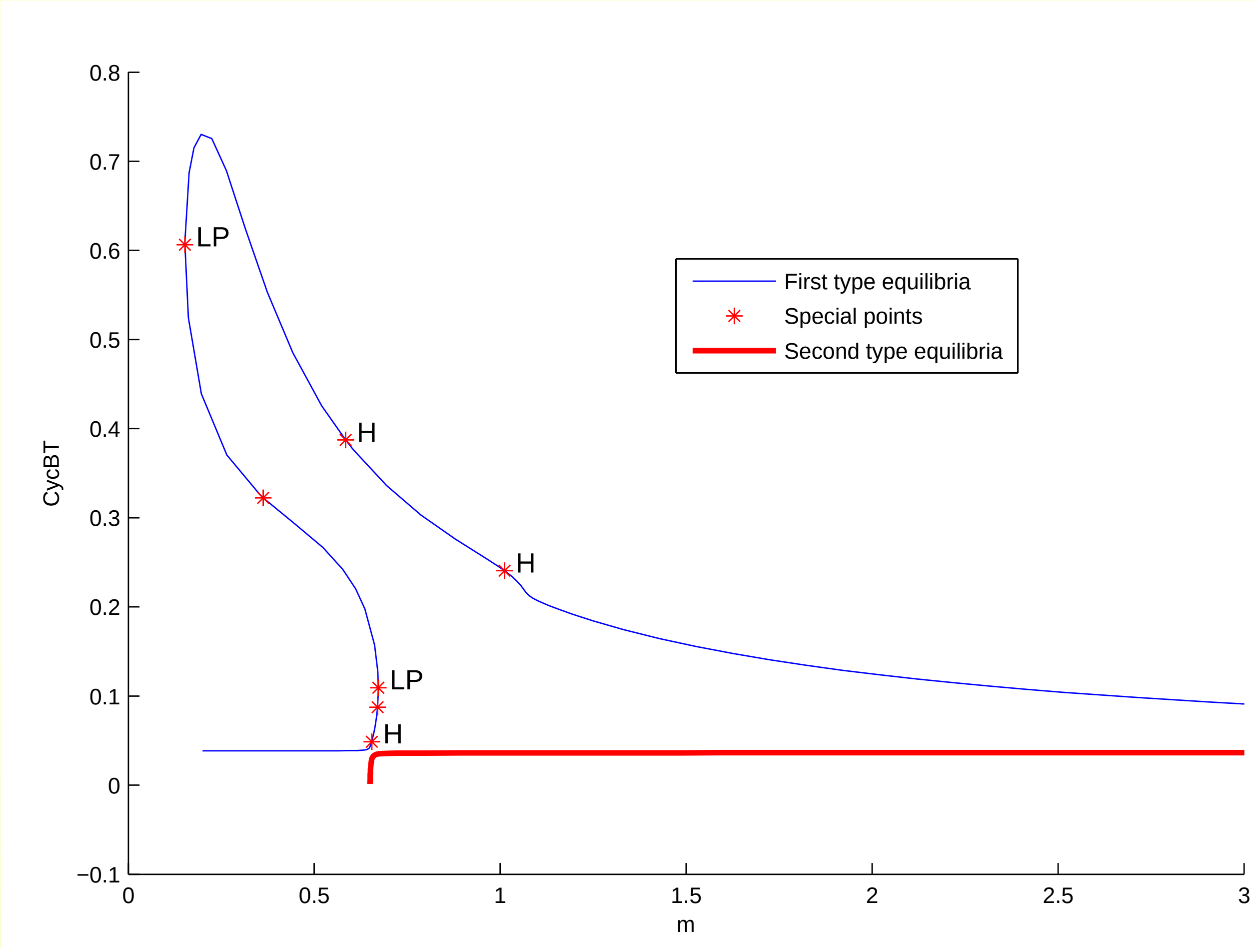
**Additional dynamic equation for a growing cell**

$$\frac{dm}{dt} = \mu m(1 - \frac{m}{m_*})$$

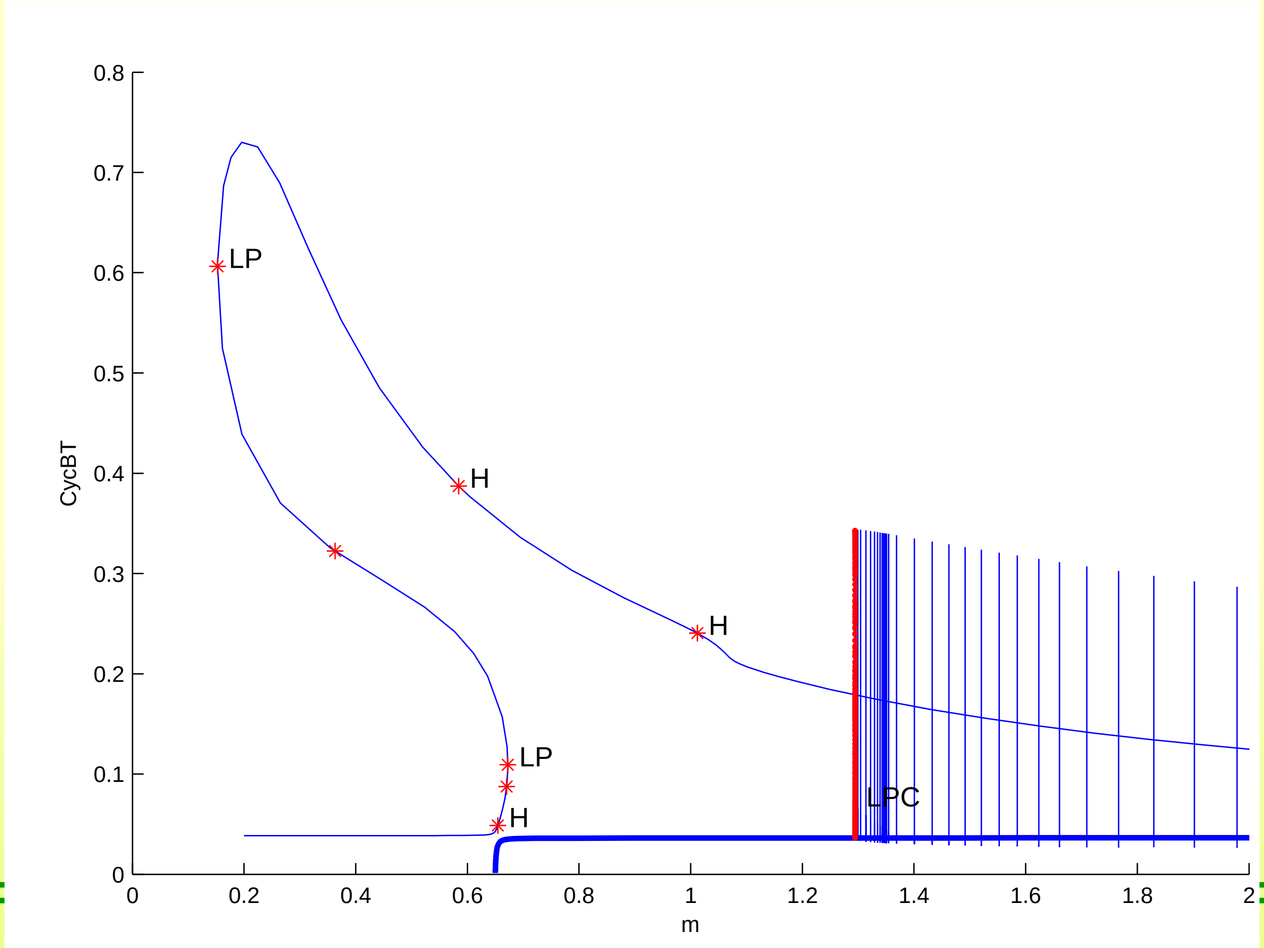
Toy model ($X$ cyclins, $Y$ APCs, $m$ and $A$ parameters):

$$\begin{cases} X' &= k_1 - (k'_2 + k''_2 Y)X \\ Y' &= \frac{(k'_3 + k''_3 A)(1-Y)}{J_3 + 1 - Y} - \frac{k_4 m X Y}{J_4 + Y}, \end{cases}$$

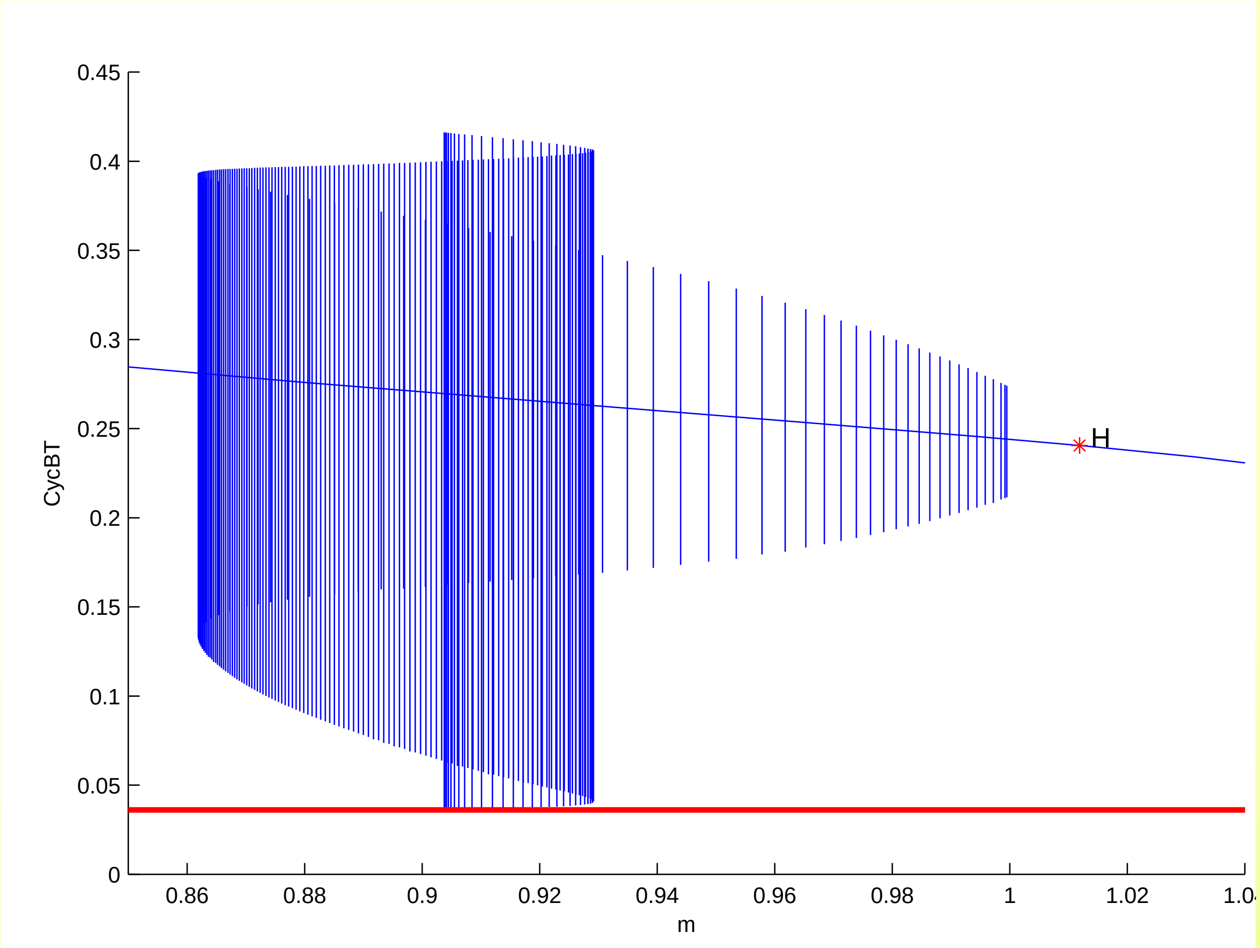
Steady states for budding yeast with cell mass m as a parameter.



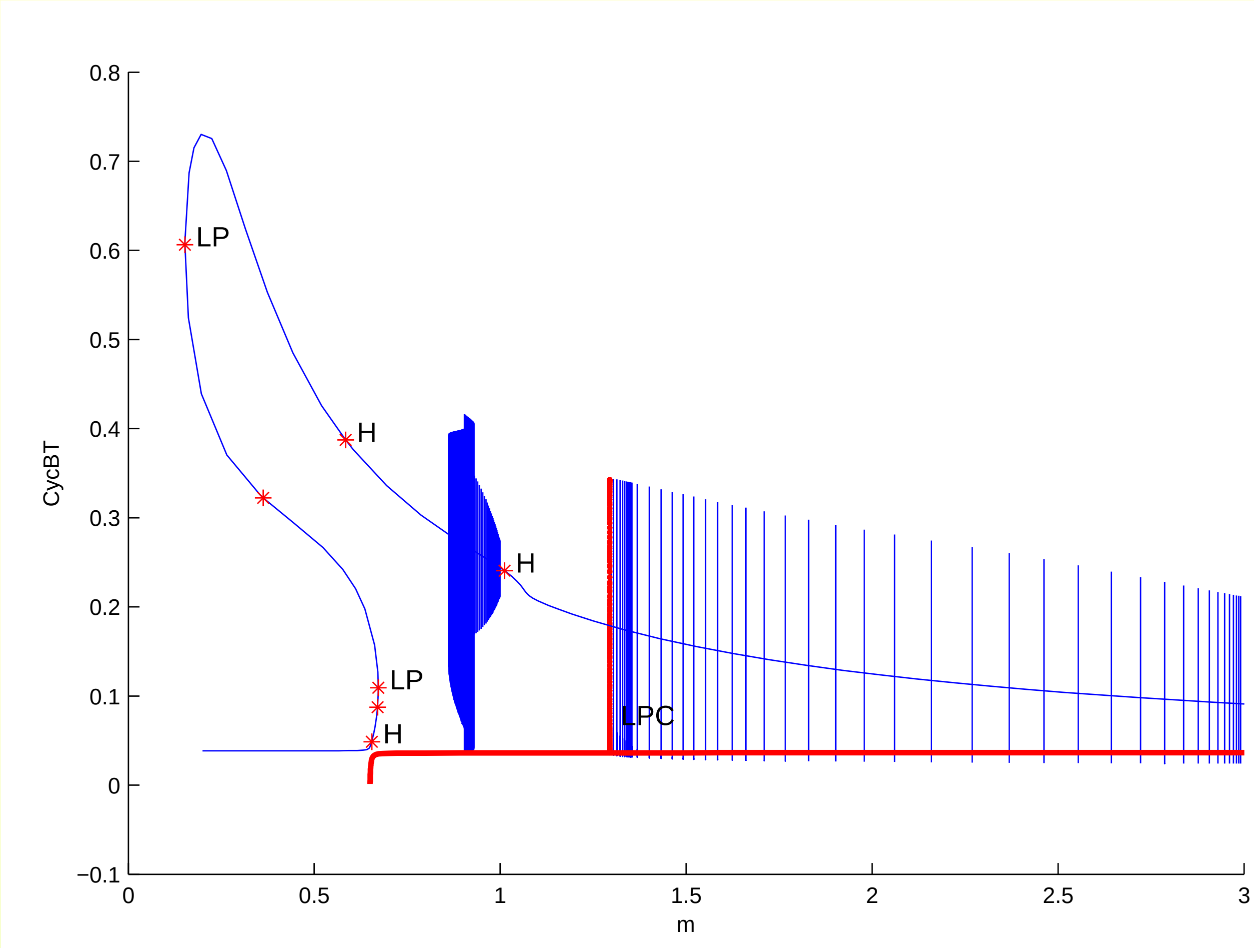
Steady states and periodic orbits for budding yeast with cell mass m as a parameter.



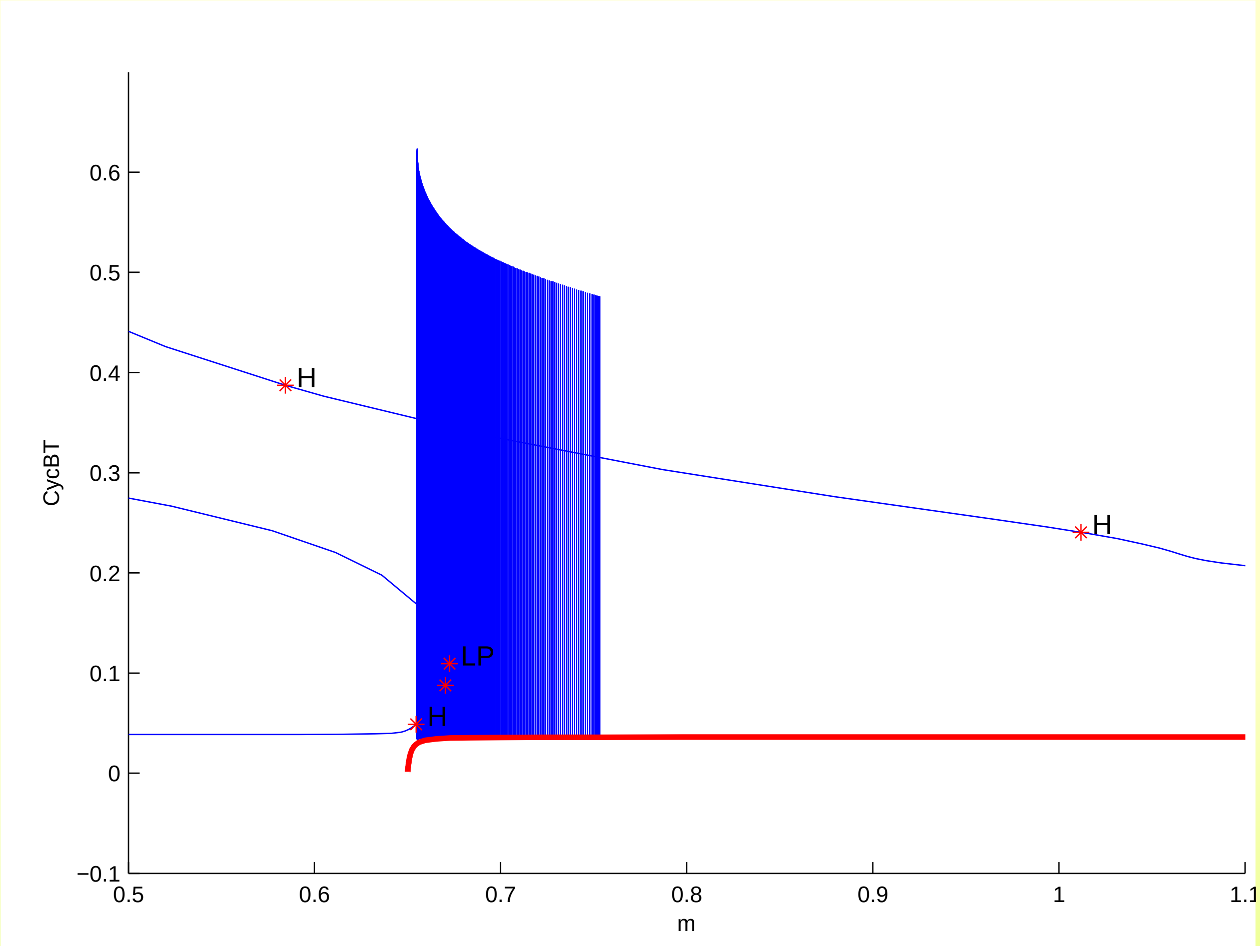
Periodic orbits coming close to a stable second type equilibrium.



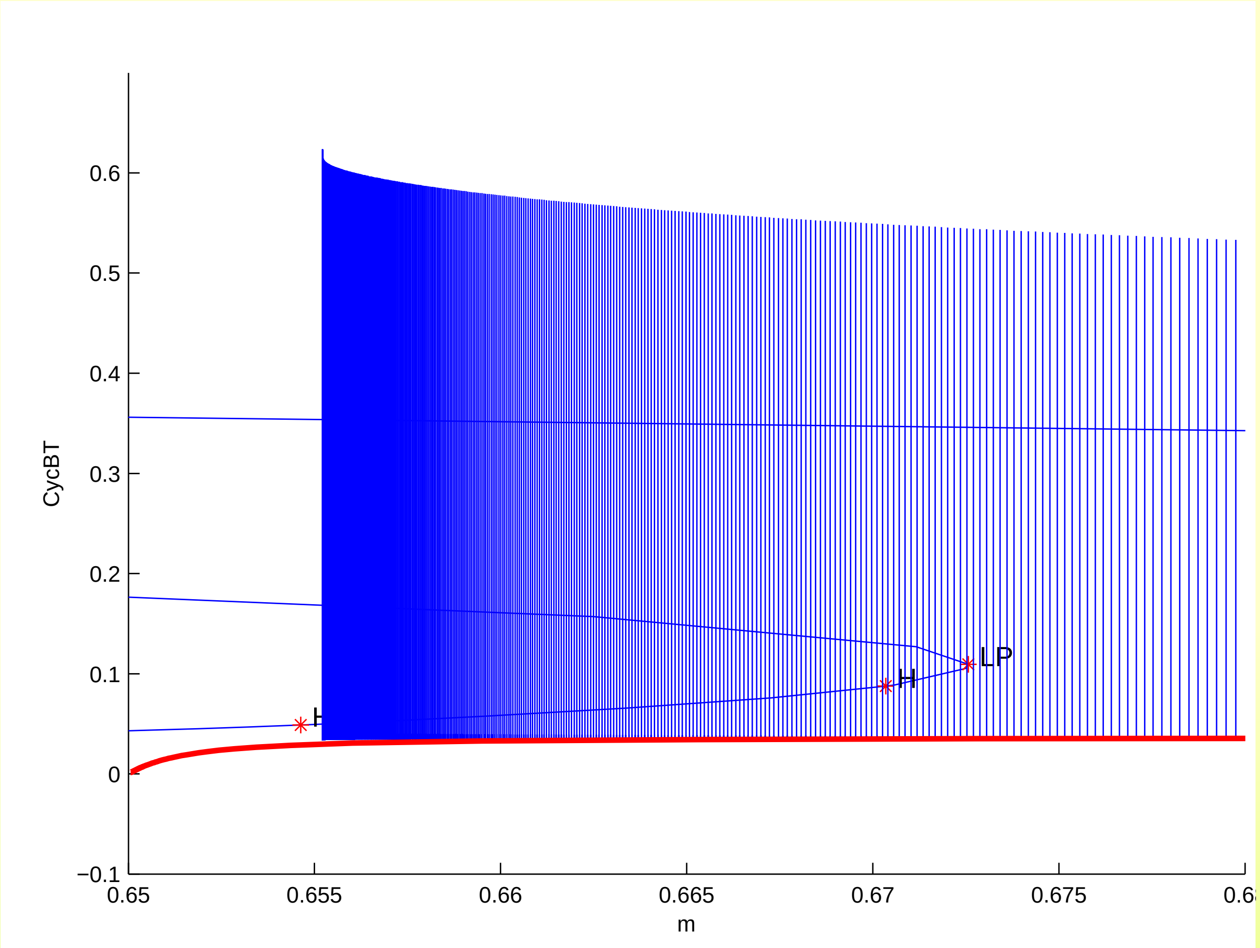
A more global picture.



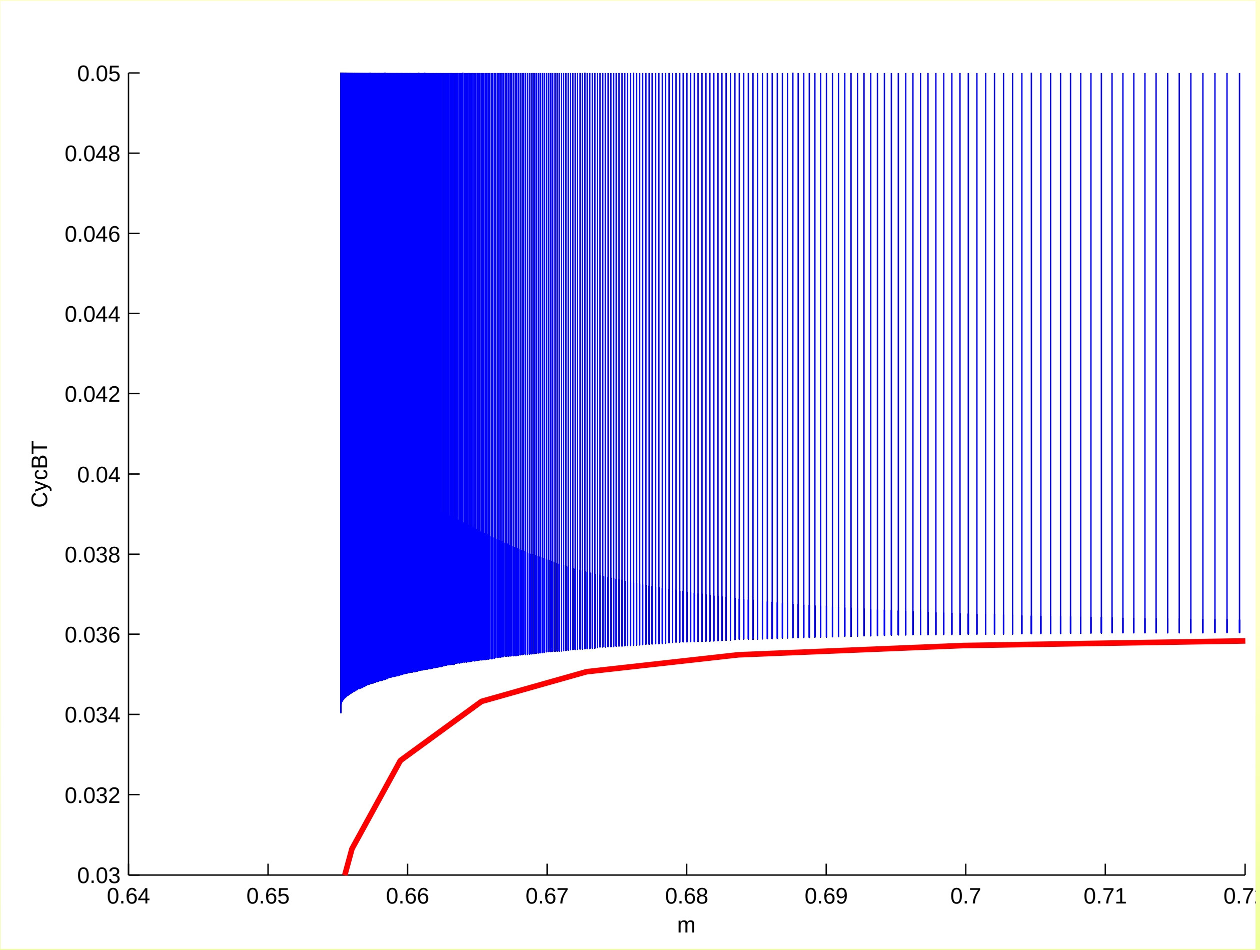
Another curve of periodic orbits.



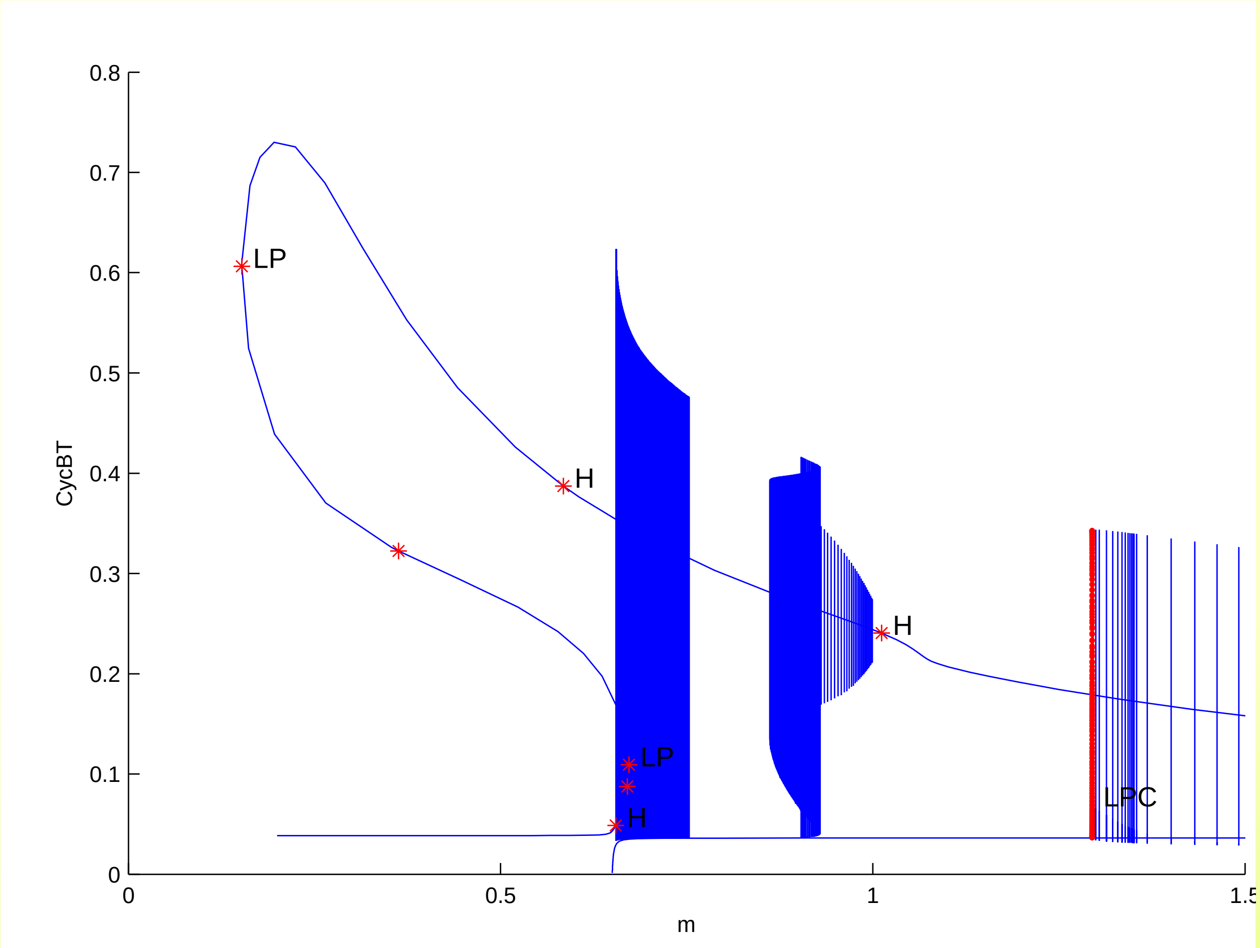
A zoom.



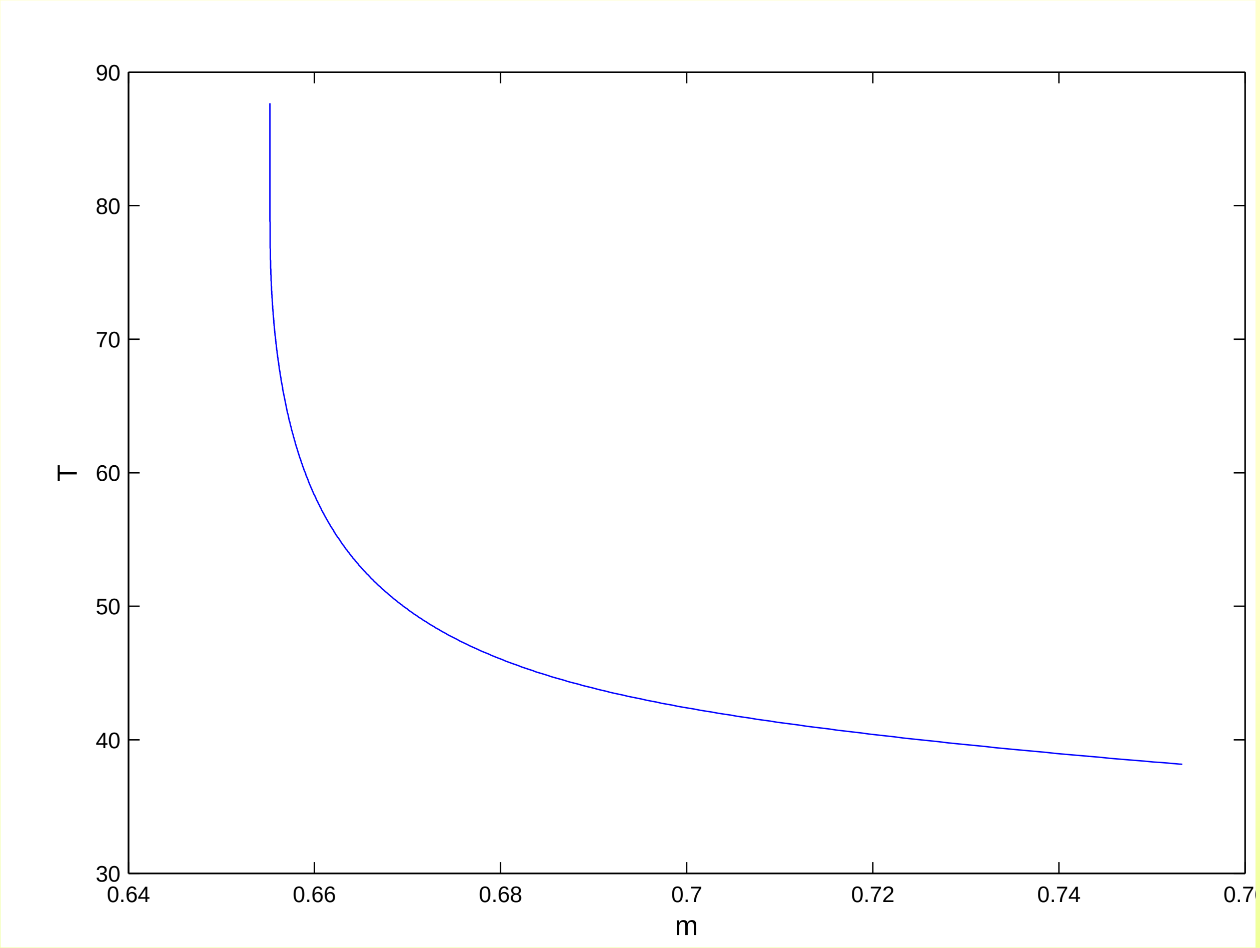
A further zoom.



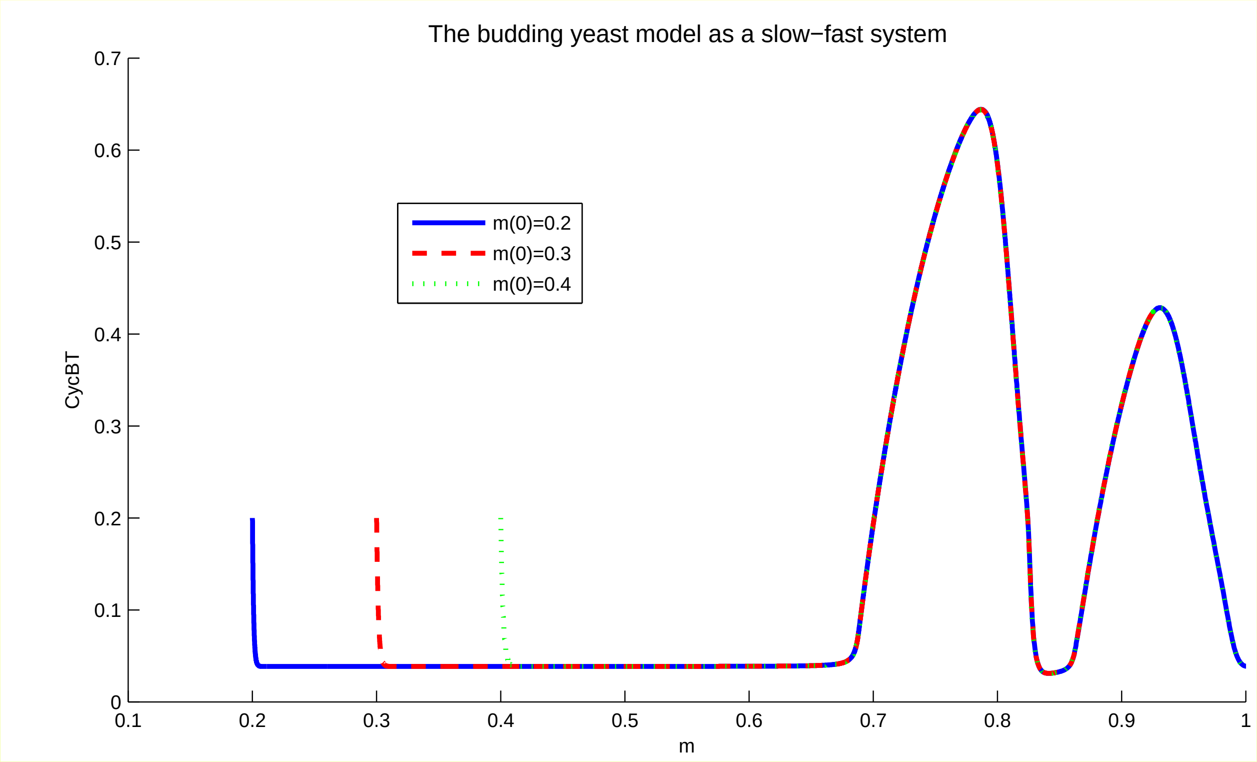
A global picture.



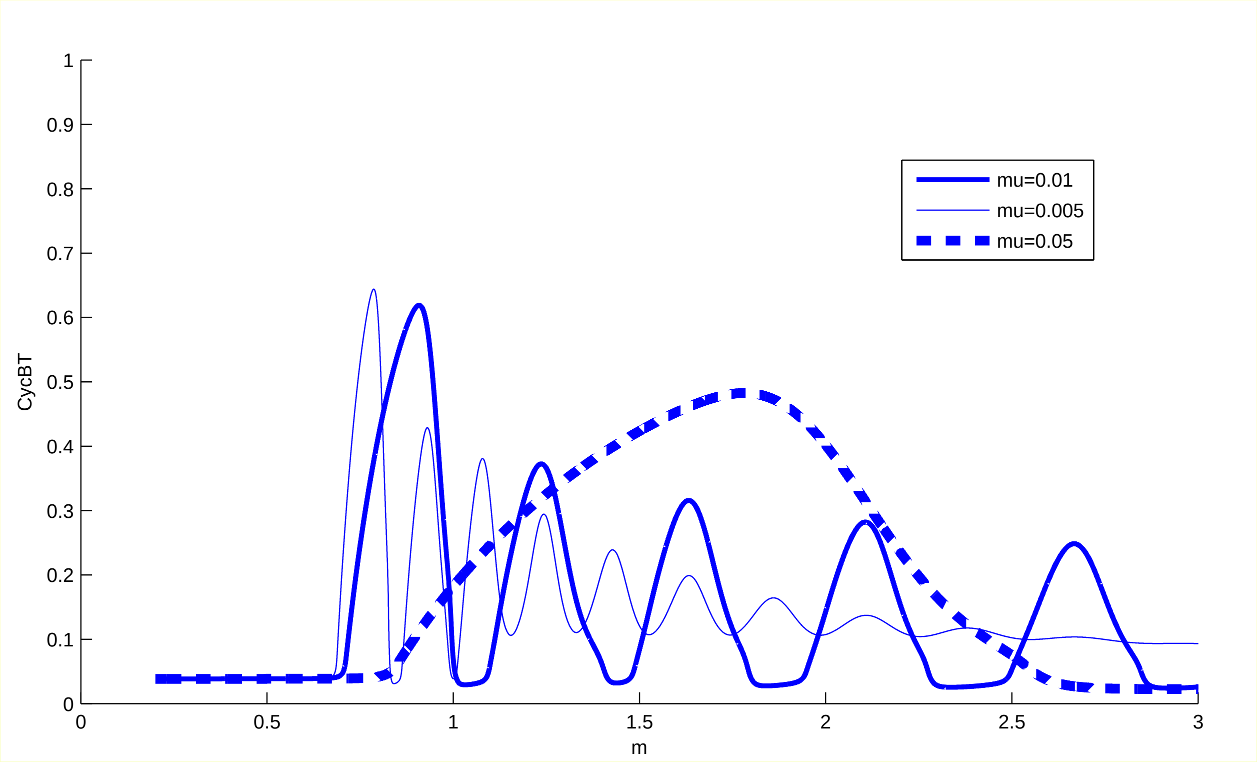
Orbit period T as a function f of cell mass m.



The budding yeast model with $m$ as a state variable and $\mu = 0.005$.



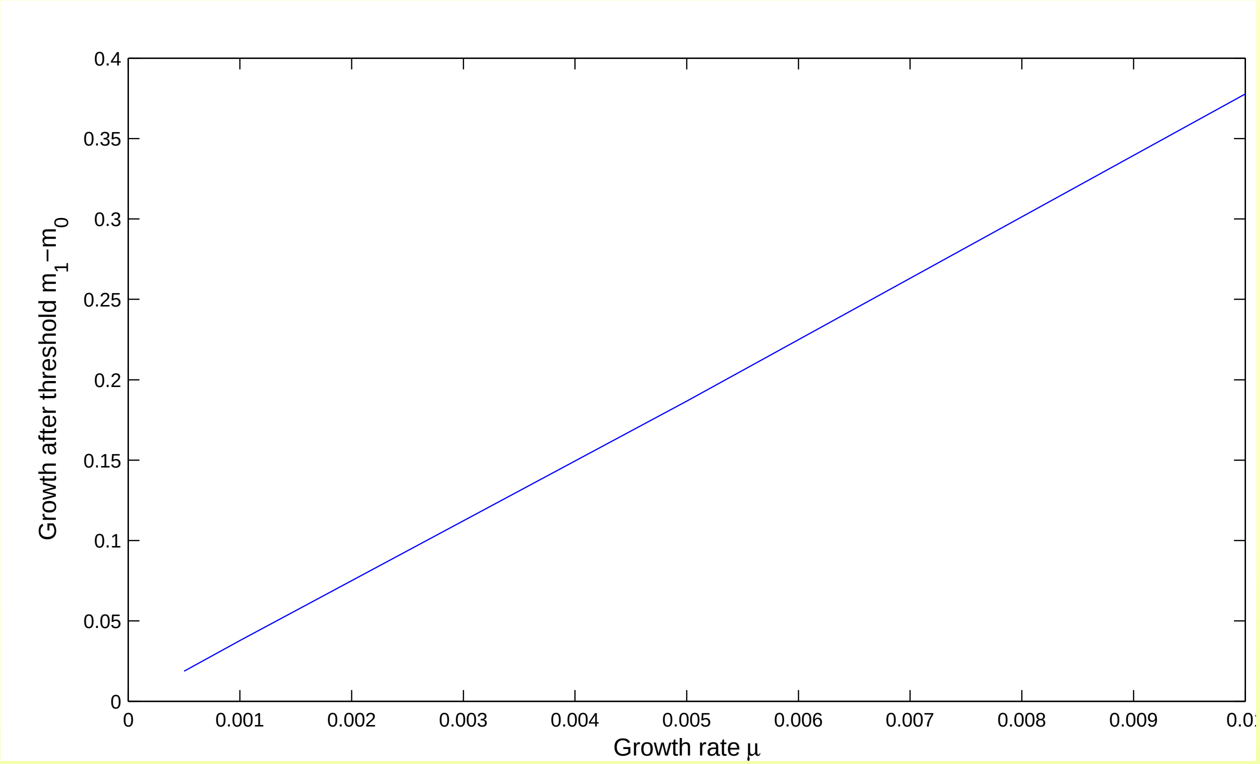
Orbits for budding yeast with cell mass m as a state variable.



Threshold of stable equilibria (Hopf point) : $m_0 = 0.6546301$.
Growth rates versus mass at division (first local minimum of $[CycB]_T$):

| | | | | |
|---|---|---|---|---|
| $\mu$ | 0.01 | 0.005 | 0.001 | 0.0005 |
| $m_1$ | 1.032418 | 0.841323 | 0.692408 | 0.673357 |
| $m_1 - m_0$ | 0.377788 | 0.186693 | 0.0377779 | 0.018727 |

Growth of a cel after replication threshold as a function of growth rate.



1. The cell cycle system behaves as a slow-fast system with cell mass as a slow variable.
2. Cell mass $m_0$ at onset of DNA-replication is a critical factor.
3. Growth of cell mass between onset of DNA-replication ($m_0$) and cell division ($m_1$) is proportional with the growth rate $\mu$, so $m_1 - m_0 \approx C\mu$ for a constant $C$ (reasons have to be studied further).
4. The time $\Delta t$ between onset of DNA - replication and cell division is given by $m_1 \approx m_0 e^{\mu \Delta t}$, i.e.

$$\Delta t = \frac{\ln \frac{m_1}{m_0}}{\mu} = \frac{1}{\mu} \ln(1 + \frac{C}{m_0}\mu) \approx \frac{C}{m_0} - \frac{1}{2}\frac{C^2}{m_0^2}\mu$$

for small values of $\mu$. Rigorously, $\Delta t$ is a decreasing function of $\mu$ and

$$\lim_{\mu \to 0} \Delta t = \frac{C}{m_0}$$

5. The full cell division cycle is a boundary value problem with a limited range of solutions for each given $\mu$. The solution is unique if a critical state variable value determines time of division.

6. If $\mu$ is a free parameter, then the solutions are determined uniquely by the cell mass at time of division. This cell mass can be chosen (approximately) between $m_0$ and $2m_0$.

7. The duration of the cell cycle is (approximately) given by

$$\frac{\ln \frac{m_1}{m_0}}{\mu} + \frac{\ln \frac{2m_0}{m_1}}{\mu} = \frac{\ln 2}{\mu},$$

where the first term is the time before DNA-replication and the second term is the time after it.

# THE END