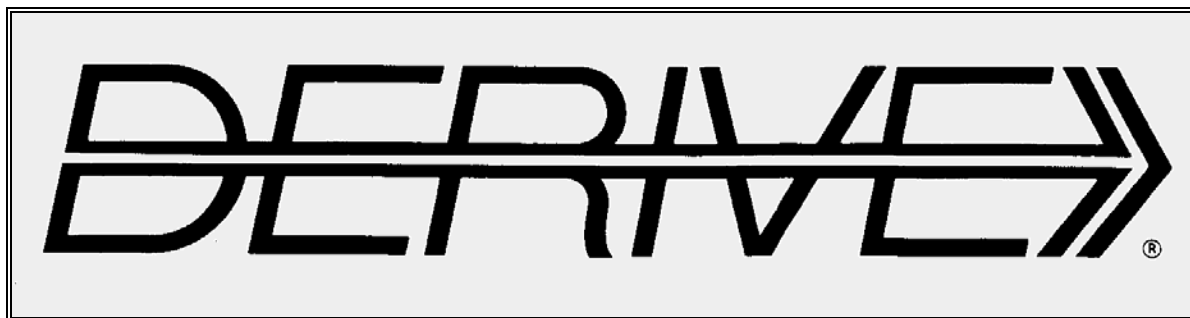


**THE BULLETIN OF THE**



**USER GROUP**

**+ CAS-TI**

**C o n t e n t s :**

- |    |   |
|----|---|
| 1  | Letter of the Editor                                    |
| 2  | Editorial - Preview                                     |
| 3  | User Forum  |
| 5  | User Forum (Conditional Sorting)                        |
| 7  | User Forum (Shaded Riemann Sums)                        |
| 8  | A new Operating System for the CAS-TIs<br>Colin Kennedy |
| 10 | Weak Acid Speciation and Titrations<br>Josef Böhm       |
| 21 | Comments on the Differential Matrix<br>Benno Grabinger  |
| 22 | A Mathematical Potato Chip                              |
| 39 | Chemistry on the CAS-TI                                 |
| 40 | From Multivalued Constants to Infinite Sets, A Report   |
| 52 | Call for Ideas  |

# DES-TIME-2006

## Dresden International Symposium on Technology and its Integration into Mathematics Education

This symposium combines 2 conferences:  
**9th ACDCA Summer Academy**  
 and  
**7th Int'l Derive and TI-CAS Conference**

**20.-23. July 2006**

**Dresden, Saxony, Germany**

### 1st Announcement and Call for Submissions

**Deadline: 15.02.2006**

#### 9th ACDCA Summer Academy

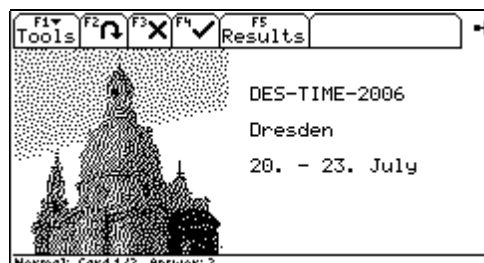
Topics include:

- CAS-based curricula and teaching methods
- Dynamic geometry systems as teaching tools
- Assessing with technology
- Internet as a teaching tool
- Dispensable and indispensable mathematical skills and abilities
- CAS as pedagogical tools for visualization, experimentation, concentration ...
- New classroom examples using CAS
- the next generation: PeCAS=Pedagogical CAS
- ...

#### 7th Derive & TI-CAS Conference

Topics include:

- Applications in mathematics, the natural sciences, research & development, economy, social sciences, industry, ...
- Interfaces to other programs / tools
- Creating and using utility files
- Programming in DERIVE, in TI-89 / Voyage 200, and in other CAS
- Problems and limitations
- Web-Resources - going online
- Teaching and Learning with DERIVE and other CAS
- Connecting computer algebra and computational logic
- ...



Please indicate

- what type of presentation you want to make (lecture, workshop, ...)
- for which of the two conference strands you intend the submission and submit using the respective form at the conference website ([www.des-time-2006.de](http://www.des-time-2006.de))

Dear DUG-Members,

I am sorry for the delay in delivering DNL#59. It was and still is a very busy time for a retired person. I had the intention to produce a smaller DNL this time and make a large issue by the end of this year – it will be DNL#60. But as you can see, it is again 52 pages. We have two very extended contributions – one of them from the fields of chemistry. I am very grateful to Dr Colin Kennedy for his article and especially for his interesting theoretic explanation. It's great to have a real life CAS-application in this DNL. I could not resist to transfer "Weak Acid Speciation to the TI". By reproducing the functions on the handheld I came across the "Differential Matrix". Ever heard about this (DERIVE function DIF\_DATA)? Find a comment on it on page 21.

The second "real life application" is modeling a Pringle, which seems to be a canned computer designed potato chip. Benno Grabinger demonstrates how motivated students can be if they are facing mathematics in their environment.

The full German version of Benno's article can be downloaded in the next future from [www.austromath.at/t3](http://www.austromath.at/t3). It is not too difficult to use the TIs for this project.

I received many requests for the User Forum. You can find two of them together with the answers which might be of interest for you. The remaining questions and answers will be published in the next DNL. As a reaction on the demand to produce contour plots in DERIVE and incited by my attempts Peter Schofield provided a wonderful package which will be presented in the "Christmas-issue".

Furthermore I am proud to announce a contribution from Johann Wiesenbauer who taught DERIVE to solve SUDOKUs. I must admit that I didn't know what SUDOKUs are when he informed me about this DERIVE challenge. He is busy providing an extremely user friendly program.

Hubert Weller sent an article about "Mathematics and Design" and last but not least I received a paper from Bernhard Kutzler "Numerics versus Symbolics".

I collected a sequence of emails which show how CAS-people are thinking and discussing about possible developments of a program. Albert Rich wrote:

*It brings to light how my thinking evolves in response to the dictates of mathematics and the comments of real world users like David's Stoutemyer and Jeffrey.*

I am very grateful for the permission to publish this exciting exchange of mails (pages 40 to 52). You are friendly invited to join the party if you have any comments or proposals on this objective. The discussion has not come to an end!

Philip Yorke from England asks for your assistance to set up a DERIVE competition (last page). If you have any ideas then please contact him directly. It would be great to inform me, too.

Finally I'd like to remind you on our next DERIVE Conference **DES-TIME 2006 in Dresden**. We are very proud that we can announce four excellent keynote speakers:

Prof. Kristina Reiss, Augsburg, Germany

Prof. Willibald Dörfler, Klagenfurt, Austria

Prof. Carl L. Leinbach, Gettysburg, USA

Dr. Terence Etchells, Liverpool

On the first page you can find the **First Call for Submissions**. It would be wonderful to meet you in Saxony next summer. Please encourage your colleagues to attend the Conference and to submit papers. All contributions are very welcome.

It remains for me to wish you a fine autumn. I hope that you like this issue and I would be happy to meet you again in December with DNL#60.



**Download all DNL-DERIVE- and TI-files from**

<http://www.austromath.at/dug/>

<http://www.bk-teachware.com/main.asp?session=375059>

The *DERIVE-NEWSLETTER* is the Bulletin of the *DERIVE & CAS-TI User Group*. It is published at least four times a year with a contents of 44 pages minimum. The goals of the *DNL* are to enable the exchange of experiences made with *DERIVE* and the *TI-89/92/Titanium/Voyage 200* as well as to create a group to discuss the possibilities of new methodical and didactical manners in teaching mathematics.

As many of the *DERIVE* Users are also using the *CAS-TIs* the *DNL* tries to combine the applications of these modern technologies.

### Contributions:

Please send all contributions to the Editor. Non-English speakers are encouraged to write their contributions in English to reinforce the international touch of the *DNL*. It must be said, though, that non-English articles will be warmly welcomed nonetheless. Your contributions will be edited but not assessed. By submitting articles the author gives his consent for reprinting it in the *DNL*. The more contributions you will send, the more lively and richer in contents the *DERIVE & CAS-TI Newsletter* will be.

Editor: Mag. Josef Böhm  
A-3042 Würmla  
D'Lust 1  
Austria  
Phone/FAX: 43-(0)2275/8207  
e-mail: nojo.boehm@pgv.at

Next issue: December 2005  
Deadline 15 November 2005

### **Preview: Contributions waiting to be published (selection)**

Two Stage Least Squares, M. R. Phillips, USA  
Some simulations of Random Experiments, J. Böhm, AUT & L. Kopp, GER  
Wonderful World of Pedal Curves, J. Böhm  
Another Task for End Examination, J. Lechner, AUT  
Tools for 3D-Problems, P. Lüke-Rosendahl, GER  
ANOVA with *DERIVE & TI*, M. R. Phillips, USA  
Hill-Encryption, J. Böhm  
Farey Sequences on the *TI*, M. Lesmes-Acosta, COL  
Simulating a Graphing Calculator in *DERIVE*, J. Böhm, AUT  
Henon & Co, J. Böhm  
Challenges from Fermat, Bj. Felsager, DEN  
Are all Bodies falling equally fast, J. Lechner, AUT  
Modelling Traffic Density, Th. Himmelbauer, AUT  
Do you know this? Cabri & CAS on PC and Handheld, W. Wegscheider, AUT  
An Interesting Problem with a Triangle, P. Lüke-Rosendahl, GER  
Diophantine Polynomials, Duncan McDougall, CAN  
Rosettes, J. Lechner, AUT  
Jacobi-Iteration, Heinz Rainer Geyer, GER  
Numerics versus Symbolics, Bernhard Kutzler, AUT  
Mathematics & Design, Hubert Weller, GER  
Implicit Plots 2D & 3D, Peter Schofield, UK  
  
and Setif, FRA; Vermeylen, BEL; Leinbach, USA; Koller, AUT; Baumann, GER;  
Keunecke, GER, .....and others

Impressum:  
Medieninhaber: *DERIVE* User Group, A-3042 Würmla, D'Lust 1, AUSTRIA  
Richtung: Fachzeitschrift  
Herausgeber: Mag. Josef Böhm  
Herstellung: Selbstverlag

**Michel Beaudin, Montreal**

I am writing this email from Nara, Japan (see the Education session of ACA-05 at <http://www.jssac.org/Conference/ACA/>). This is my first trip to Japan: beautiful country, rich history and culture! My wife loves it!

In my talk, I used Derive 6 slider bars for 2D and 3D animation (I should repeat this next year in Dresden). But when I used the "show step simplification", the audience could not believe it! I used the (known) example of  $\int \frac{1}{1+\tan(x)^r} dx$  whose value is  $\pi/4$  for any value of the parameter  $r$ .

I also used the TI Voyage 200 in order to show that, if we want computer algebra to be used by students, we must do it each day, in the classroom, not only in computer labs.

I think that it is very important to show Derive and the TI outside Derive and TI conferences. ACA conferences should not be only Maple and Mathematica oriented ...

Best regards and see you next year in Dresden, Germany!

Michel Beaudin

**Don Phillips, Washington**

Josef,

Hello! Hope you've had a good summer!. I've been busy. I put my actuarial math programs from Derive into the TI-89. Everything seems to be working as it should. Once I get a write up for how to use it, I'll send it on to you. I'm also almost done with putting my bond programs onto the TI-89, calendar functions and all. I'll send that onto you also when I write up some instructions.

Regards,

Don

...it is unworthy of excellent men to lose hours like slaves  
in the labor of computation.

Gottfried Wilhelm Leibniz

**Manuel Garcia Mateos, Germany**

.....

2. Preparing a teaching project to the subject „Complex Numbers“ I used DNL#55 to among others produce a geometric representation of the fundamental theorem. In this DNL I saw that a contour plot was made by means of the TI.

My question is how to project the contour plot into the xy-plane in order to show the graph  $\text{mod\_surf}(f)$  in a 3D-Plot Window? It would be great to implement a contour plot command (function) as a package in DERIVE (similar to MAPLE).

Many thanks

Manuel Garcia Mateos.

**DN.:** Manuel sent a great article – translated into English – which I forwarded to Bert Waits and Frank Demana, who were the authors of the contribution in DNL#55. This article will be published in one of the next DNLs, many thanks Garcia. Bert immediately wrote back:

Many, many thanks for 1. Publishing the paper and 2. sending this email! It is great, as you noted, to see ones work recognized so nicely!

Bert

But this was not the whole reaction. I sent the request for contour plots to Albert Rich and Theresa Shelby who involved our friend Peter Schofield.

In the meanwhile the problem how to provide contour plots made me busy and after some attempts I found a way to produce them and sent my first results to Garcia, Albert and Peter (see the plots on the bottom of this page). This is Albert's answer:

Very clever. Looks like it has the makings of a nice article for the Newsletter.

Peter studied my function and it made him busy, too:

Hello Josef, Albert and Theresa,

Thank you for your email Josef.

Your contour plotting idea is extremely interesting. It reminds me of the glass ornament cubes (you see in shops) with the lazar etched figures inside. The finished 3D-plots are very attractive.

The calculations appear to take a long time. I am attaching two Derive files that may suggest ways of speeding them up.

#### 1. polycontour.mth

Calculation times can sometimes be reduced by restricting your function input to implicit polynomial equations of form  $f(x,y,z) = 0$ . This file also contains another way of persuading Derive to 3D-plot small points. (It still takes the 3D-plot widow around 60 seconds to process some of the large data lists.)

#### 2. Nsolutions.mth

This file contains a faster procedure (N\_SOLUTIONS) for finding all of the zeros of  $f(x) = 0$  for  $a \leq x \leq b$ . I'm not entirely happy with it, but it works. Perhaps Albert can improve it?

In fact, something like this for both NSOLUTIONS and NSOLVE would be a useful addition to a future update of Derive 6.

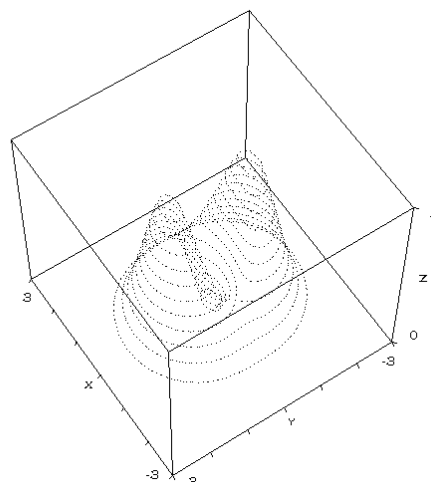
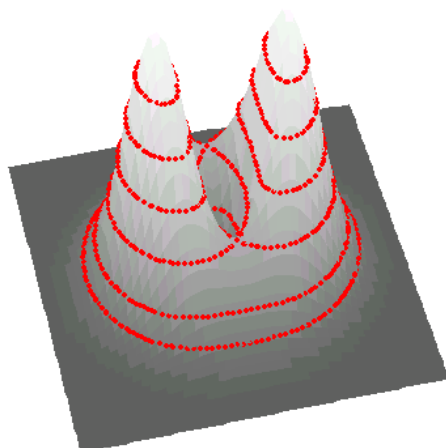
Can I request it for the wish list?

Hope this helps.

Best wishes,

Peter

Here are two results of contour plots. Another one is given at the end of the "Pringles" contribution. We will come back with the full contribution in the next DNL!



Dirk Callens, Belgium

Hi Josef,

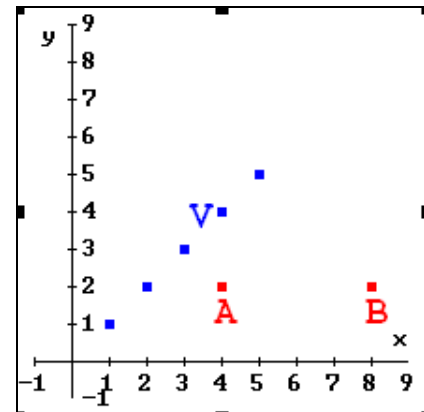
I have a little problem with Derive. (See attachment). Can you solve it ?

Thanks, Dirk

#1: CaseMode := Sensitive

We have two points A and B, and a vector V of points in the plane.

$$\#2: \left[ A := [4, 2], B := [8, 2], V := \begin{bmatrix} 1 & 1 \\ 2 & 2 \\ 3 & 3 \\ 4 & 4 \\ 5 & 5 \end{bmatrix} \right]$$



a) Sort the vector V in descending order of the distance to the origine.

$$\#3: \text{SORT}(V, |X| \geq |Y|, X, Y) = \begin{bmatrix} 5 & 5 \\ 4 & 4 \\ 3 & 3 \\ 2 & 2 \\ 1 & 1 \end{bmatrix}$$

Ok, this works.

b) Sort the vector V in descending order of the distance to the point [2,2] and

c) Sort the vector V in descending order of the surface of the triangle with vertices A, B and a point of V.

$$\#4: \text{SORT}(V, |X - [2, 2]| \geq |Y - [2, 2]|, X, Y) = \text{SORT} \left( \begin{bmatrix} 1 & 1 \\ 2 & 2 \\ 3 & 3 \\ 4 & 4 \\ 5 & 5 \end{bmatrix}, ?, X, Y \right)$$

$$\#5: \text{SORT} \left( V, \frac{1}{2} \cdot |\text{CROSS}(X - A, X - B)| \geq \frac{1}{2} \cdot |\text{CROSS}(Y - A, Y - B)|, X, Y \right) = \text{SORT} \left( \begin{bmatrix} 1 & 1 \\ 2 & 2 \\ 3 & 3 \\ 4 & 4 \\ 5 & 5 \end{bmatrix}, ?, X, Y \right)$$

Why don't both procedures work ??

Although we can make a table of the areas !!

$$\#6: \text{TABLE} \left( \frac{1}{2} \cdot |\text{CROSS}(X - A, X - B)|, X, V \right) = \begin{bmatrix} [1, 1] & 2 \\ [2, 2] & 0 \\ [3, 3] & 2 \\ [4, 4] & 4 \\ [5, 5] & 6 \end{bmatrix}$$

I sent Dirk's request to Johann Wiesenbauer, our "Titbits Man" and he very quick found a satisfying solution for sorting under conditions:

## Conditional Sorting with Derive

As for conditional sorting, it is rather the rule than the exception that Derive can't cope with user defined comparisons.

It can though, with "a little help" of the user. The trick is to insert the expression used for the sorting as a first component and to sort this two element vector. Then the second column gives the result. This is what my function `mysort(v,a,e)` does, where `v` denotes the vector to be sorted, `e` is the expression that is used for sorting and `a` stands for sorting "in ascending order".

```
mysort(v, e := x, a := true) :=
  If a
#1: (SORT(VECTOR([SUBST(e, x, v_), v_], v_, v))) COL 2
    REVERSE(mysort(v, e))
```

$$\#2: \text{mysort}(v, |x - [2, 2]|, \text{false}) = \begin{bmatrix} 5 & 5 \\ 4 & 4 \\ 3 & 3 \\ 1 & 1 \\ 2 & 2 \end{bmatrix}$$

$$\#3: \text{mysort}\left(v, \frac{1}{2} \cdot |\text{CROSS}(x - A, x - B)|\right) = \begin{bmatrix} 2 & 2 \\ 1 & 1 \\ 3 & 3 \\ 4 & 4 \\ 5 & 5 \end{bmatrix}$$

$$\#4: \text{mysort}\left(v, \frac{1}{2} \cdot |\text{CROSS}(x - A, x - B)|, \text{false}\right) = \begin{bmatrix} 5 & 5 \\ 4 & 4 \\ 3 & 3 \\ 1 & 1 \\ 2 & 2 \end{bmatrix}$$

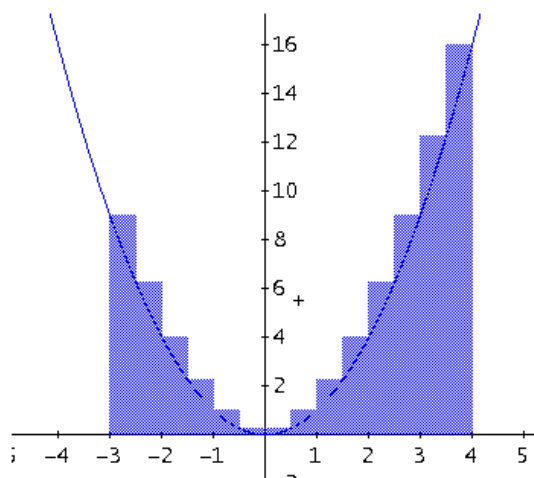
## Shaded Riemann Sums

Martin Wuttke from Berlin wrote to Walter Wegscheider, the author of the DERIVE Online Tutorial ([www.austromath.at/daten/derive](http://www.austromath.at/daten/derive)) how to produce the shaded area under the step-function (Riemann sums). He tried working with **AreaUnderCurve** and found it very tiresome doing it piece by piece. Walter had produced the illustration in his tutorial using GeoGebra ([www.geogebra.at](http://www.geogebra.at)) and asked me if I had any easier way to do this with DERIVE

**DNL:** See the results of my efforts on the next page. You will find how to plot "real" upper- and lower sums in one of the next revised old DNLs.

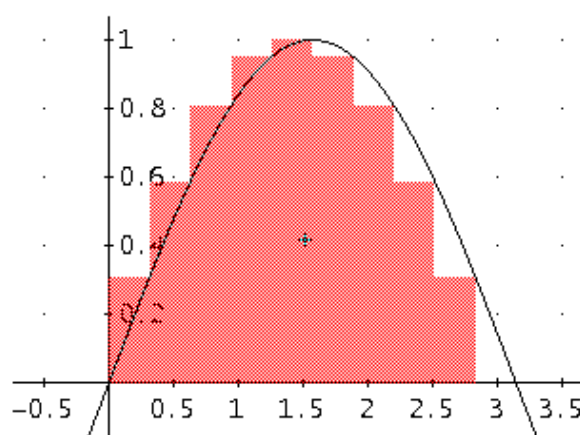


#22:  $\text{osumsh}(x^2, -3, 4, 14)$

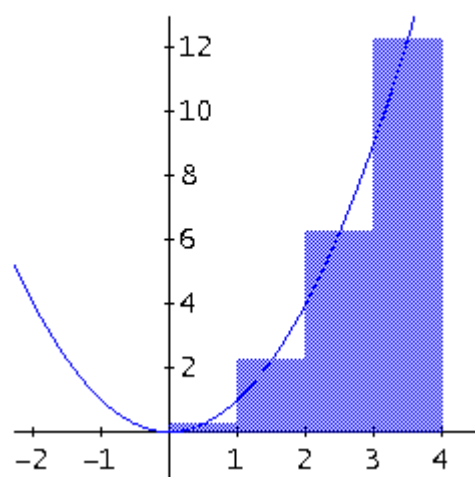


#5:  $\text{rsumsh}(\text{SIN}(x), 0, \pi, 10)$

#6:  $\text{SIN}(x)$



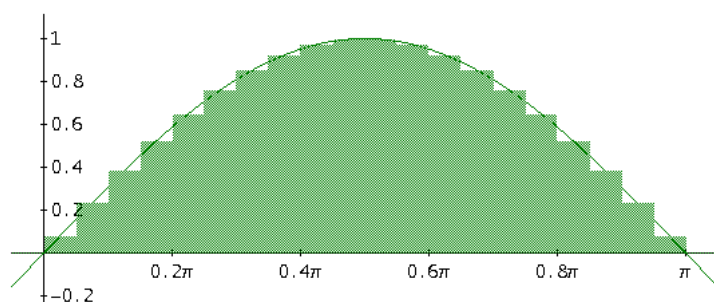
#8:  $\text{msumsh}(x^2, 0, 4, 4)$



The Functions above are small programs, the next function is one VECTOR-command involving the CHI-function:

$$\text{msumchi}(u, a, b, n) := \left[ \text{VECTOR} \left( 0 \leq y \leq \text{SUBST} \left( u, x, a + \frac{b-a}{2 \cdot n} + \frac{(b-a) \cdot (k-1)}{n} \right) \right) \cdot x \left( a + \frac{(b-a) \cdot (k-1)}{n}, x, a + \frac{k \cdot (b-a)}{n} \right), \right. \\ \left. k, 1, n \right), u \Big]$$

$\text{msumchi}(\text{SIN}(x), 0, \pi, 20)$



[riemsum.dfw](#)

Download the new Operating System for TI-CAS devices!

## Voyage200/TI-89T OS enhancements

Operating System Version 3.10

### Key New Functionality

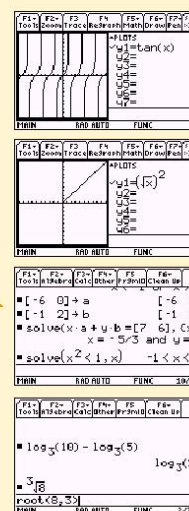
- **Graphing Improvements**
- **Domain Improvements**
- Gradian Angle Measure
- Population Standard Deviation
- Implicit derivatives
- **Increased Solving Capabilities**
- **Log to Any Base and Root Functions**
- Bug fixes

### Availability

- Available [next from the TI web site](#)
- **Available “out of the box”:** Running change, not available pre-loaded on units approx. until January 1<sup>st</sup> TI-89T, April 1<sup>st</sup> Voyage 200

### Compatibility

- Compatible with [Voyage 200](#) and [TI-89 Titanium](#)!



## Customer needs addressed

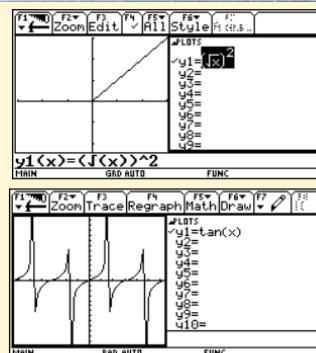
- New OS extends current CAS functionality to better meet advocates' expectations, especially those which our competitors have already addressed.
- New OS corrects some of the long-standing deficiencies of the CAS.

## Key Benefits

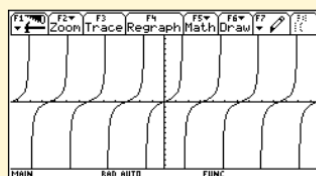
- **More Pedagogical:** Domain Tracking, Graphing and other Improvements make our solution a better then before fit for high school math environment.
- **More Powerful:** Solving Inequalities, Implicit Derivatives and Gradian Angle Measure make our CAS system more versatile.
- **Easier to Use:** Ease of use features include easier access to Apps from Desktop, better interface for solving systems of equations using vectors, root notation and log to any base functionality.
- **Support for European Audience:** The new OS is supported by updated localizer Apps and comprehensive guidebooks.

- Block auto-simplification when graphing
- Warn the user that intermediate steps in a computation involved complex numbers and when the domain may have changed
- Give the user the option to remove “faux” asymptotes by introducing a smart discontinuity detection mechanism.
- Graphing piecewise defined functions improved as well!

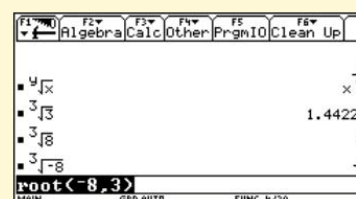
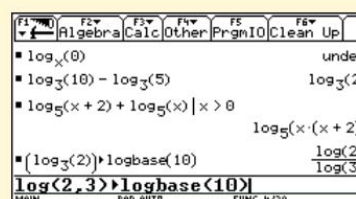
Before...



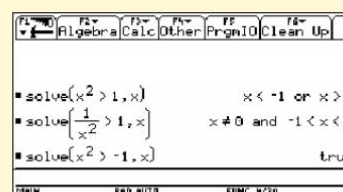
After...



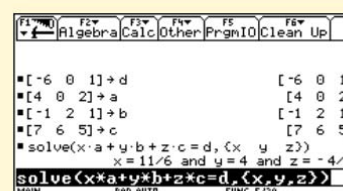
- Support for logarithm to any base including transfinite bases and algebraic expressions.
- Easier access to log functionality.
- Try to prevent the user noticeable conversion to natural log.
- Natural log and any log conversion commands.
- Subscripts.
- Support for root function



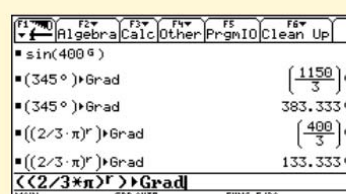
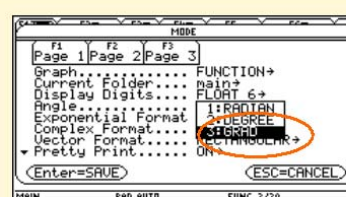
- Increase the ability of the TI CAS to cover inequalities involving polynomial, rational, and absolute value expressions.



- Solve fundamental problems involving vectors. These include decomposition and intersection problems (two lines, a line and a plane, two planes) involving parametric forms.



- Gradian angle measure has long been a request from European teachers.
- Add Gradian angle MODE.
- Add Gradian angle indicator on the status line: GRD.
- Use superscript G to indicate the new angle measure.
- Implement ► GRAD for angle conversion.



## Weak Acid Speciation and Titrations with Derive

Colin Kennedy, UK

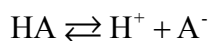
### Abstract

I present an account of the use of **Derive 6** in creating speciation and titration graphs for weak mono-, di- and triprotic acids in solutions of known ionic strength. The speciation of weak acids depends on the relevant acid dissociation constants, the solution pH, the temperature, and the ionic strength of the solution. Dissociation constants are often obtained using pH titrations, but there are many pitfalls when it comes to interpreting the results, even supposing the titration has been carried out with some care! It is hoped that the methods outlined here will encourage students towards a more thoughtful approach to some aspect of solution chemistry.

This account divides into (1), an outline of the relevant maths and **Derive** functions relating to the solution chemistry involved, (2), a **Derive utility file** with brief descriptions of the authored expressions, and (3), some examples, using **Derive**, of the speciation and titration graphs that can be created.

### (1) An outline of the maths and Derive functions.

A quantitative description of the dissociation of a weak acid requires knowledge of all the relevant dissociation constants. Ignoring the solvent water, a monoprotic weak acid dissociates as-



The ‘thermodynamic’ dissociation constant, here called  $K_t$ , is written in terms of the activities, the ‘a’ terms below. These are the *effective* concentrations of the ions, etc, so that  $K_t$  does not change with the ionic strength of the solution, although it can change with temperature.

$$K_t = \frac{a_{\text{H}^+} a_{\text{A}^-}}{a_{\text{HA}}} \quad (1)$$

Each activity may be written as the product of a molar concentration times an activity coefficient. For example, for hydrogen ions,

$$a_{\text{H}^+} = [\text{H}^+] \gamma_{\text{H}^+} \quad (2)$$

Using this definition of activity for HA and A<sup>-</sup> in (1),

$$K_t = \frac{a_{\text{H}^+} [\text{A}^-]}{[\text{HA}]} \times \frac{\gamma_{\text{A}^-}}{\gamma_{\text{HA}}} = K_m \times \frac{\gamma_{\text{A}^-}}{\gamma_{\text{HA}}} \quad (3)$$

-where  $K_m$  is the ‘mixed’ acid dissociation constant. Mixed acid dissociation constants are appropriate for use with both speciation and titration graphs as hydrogen ions are usually measured using a pH electrode calibrated to the log of H<sup>+</sup> ion *activity*, where  $\text{pH} = -\log_{10}(a_{\text{H}^+})$ , whereas other ions present are assessed as their *molar concentrations*.

To calculate the activity coefficients, the total ionic strength,  $I$ , of the solution must be calculated where-

$$I = \frac{1}{2} \sum_i c_i z_i^2 \quad (4)$$

For each ion,  $i$ ,  $c$  and  $z$  are the molar concentration and charge respectively. When carrying out pH titrations to determine dissociation constants, a known, relatively high molar concentration of a strong electrolyte, eg, KCl, is included, so that  $I$  is determined to a reasonable accuracy by that electrolyte alone. For a single strong electrolyte,  $M_m B_b$ ,  $I$  is found using the function **SING\_SALT\_I** (line #8 in the **Derive** utility file).

Single ion activity coefficients for most common ions can be calculated to reasonable accuracy for  $I < 0.5$  using the semi-empirical Davies formula<sup>1</sup>,-

$$\log_{10} \gamma_i = -Az_i^2 \left( \frac{\sqrt{I}}{(1 + \sqrt{I})} - 0.2I \right) \quad (5)$$

This expression is the basis for the **Derive** expression **ACT\_COEF\_H** (#10 in the **Derive** utility file), which calculates the activity coefficient of  $H^+$  ions at ionic strength  $I$ .  $A$  is a parameter which varies with temperature and equals 0.51 at 298K<sup>1</sup>.

Mixed acid dissociation constants, the  $K_m$ 's, can be calculated from the thermodynamic ones, the  $K_i$ 's, by combining Equations (3) and (5). The charge,  $z$ , on the fully protonated weak acid as well as the dissociation step,  $n$ , is also needed to formulate **PKMN** (**Derive** utility file expression #12). Note that the values calculated in #12 are in 'p' form. For example, in the second dissociation step for phosphoric acid,  $H_3PO_4$ , the total charge on the *fully protonated* acid,  $z = 0$ , and the dissociation step,  $n = 2$ . So in this case **PKMN** would give  $pK_{m2} = -\log_{10}(K_{m2})$ . Note that  $z$  is not invariably zero, for example, for the diprotic acid glycine,  $z = +1$ .

### Speciation formulae

The speciation formulae for mono-, di- and triprotic acids as a function of pH are derived from the mass balance equations for the appropriate acids, and the relevant 'mixed' acid dissociation constants. ' $C_A$ ' in the formulae below is the total molar concentration of the acid.

For **monoprotic acids** these are-

$$C_A = [HA] + [A^-] \quad (6)$$

$$K_{m1} = \frac{a_{H^+} [A^-]}{[HA]} \quad (7)$$

Leading to-

$$\alpha_0 = \frac{[HA]}{C_A} = \frac{1}{\frac{K_{m1}}{a_{H^+}} + 1} \quad \text{and} \quad \alpha_1 = \frac{[A^-]}{C_A} = \frac{1}{1 + \frac{a_{H^+}}{K_{m1}}} \quad (8)$$

$\alpha_0$  and  $\alpha_1$  are the fractions of the acid that have lost 0 and 1 protons respectively. The subscript  $m1$  indicates that the dissociation constant relates to the first proton (and here, the only one) to dissociate from the fully protonated acid. The corresponding **Derive** functions are **MPA0** (#18) and **MPA1** (#19). As with all the **Derive** functions, hydrogen ion activities have been converted to pH, and all  $K_m$  have been converted to  $pK_m$ . The molar concentrations of  $[HA]$  and  $[A^-]$  are obtained by multiplying the  $\alpha_0$  and  $\alpha_1$  functions by  $C_A$ . The corresponding **Derive** functions are **HA** (#21) and **Am** (#22).

For **diprotic acids**-

$$C_A = [H_2A] + [HA^-] + [A^{2-}] \quad (9)$$

$$K_{m1} = \frac{a_{H^+} [HA^-]}{[H_2A]} \quad \text{and} \quad K_{m2} = \frac{a_{H^+} [A^{2-}]}{[HA^-]} \quad (10)$$

So,-

$$\begin{aligned} \alpha_0 &= \frac{[H_2A]}{C_A} = \frac{1}{1 + \frac{K_{m1}}{a_{H^+}} + \frac{K_{m1}K_{m2}}{a_{H^+}^2}} \\ \alpha_1 &= \frac{[HA^-]}{C_A} = \frac{1}{\frac{a_{H^+}}{K_{m1}} + 1 + \frac{K_{m2}}{a_{H^+}}} \\ \alpha_2 &= \frac{[A^{2-}]}{C_A} = \frac{1}{\frac{a_{H^+}^2}{K_{m1}K_{m2}} + \frac{a_{H^+}}{K_{m2}} + 1} \end{aligned} \quad (11)$$

$\alpha_0$ ,  $\alpha_1$  and  $\alpha_2$  are the fractions of the acid that has lost 0, 1 and 2 protons respectively. The corresponding **Derive** functions for  $\alpha_0$ ,  $\alpha_1$  and  $\alpha_2$  are **DPA0** (#27), **DPA1** (#28) and **DPA2** (#29). The molar concentrations of  $[H_2A]$ ,  $[HA^-]$  and  $[A^{2-}]$  as functions of pH are given by the **Derive** formulae **H2A** (#31), **HAm** (#32) and **Am2** (#33) respectively.

Similarly, for **triprotic acids**, the total acid concentration is-

$$C_A = [H_3A] + [H_2A^-] + [HA^{2-}] + [A^{3-}] \quad (12)$$

Where the appropriate 'mixed' acid dissociation constant formulae are  $K_{m1}$ ,  $K_{m2}$  and  $K_{m3}$ , and the fractions of the triprotic acid that has lost 0, 1, 2 and 3 protons respectively are  $\alpha_0$ ,  $\alpha_1$ ,  $\alpha_2$  and  $\alpha_3$  respectively.-

$$\begin{aligned}
\alpha_0 &= \frac{[H_3A]}{C_A} = \frac{1}{\frac{K_{m1}K_{m2}K_{m3}}{a_{H^+}^3} + \frac{K_{m1}K_{m2}}{a_{H^+}^2} + \frac{K_{m1}}{a_{H^+}} + 1} \\
\alpha_1 &= \frac{[H_2A^-]}{C_A} = \frac{1}{\frac{K_{m2}K_{m3}}{a_{H^+}^2} + \frac{K_{m2}}{a_{H^+}} + 1 + \frac{a_{H^+}}{K_{m1}}} \\
\alpha_2 &= \frac{[HA^{2-}]}{C_A} = \frac{1}{\frac{K_{m3}}{a_{H^+}} + 1 + \frac{a_{H^+}}{K_{m2}} + \frac{a_{H^+}^2}{K_{m1}K_{m2}}} \\
\alpha_3 &= \frac{[A^{3-}]}{C_A} = \frac{1}{1 + \frac{a_{H^+}}{K_{m3}} + \frac{a_{H^+}^2}{K_{m2}K_{m3}} + \frac{a_{H^+}^3}{K_{m1}K_{m2}K_{m3}}}
\end{aligned} \tag{13}$$

The corresponding **Derive** functions are **TPA0** (#38), **TPA1** (#39), **TPA2** (#40) and **TPA3** (#41) respectively. The molar concentrations of  $[H_3A]$ ,  $[H_2A^-]$ ,  $[HA^{2-}]$  and  $[A^{3-}]$  as functions of pH are given by the **Derive** formulae **H3A** (#43), **H2Am** (#44), **HAm2** (#45) and **Am3** (#46) respectively.

### Creating titration graphs using Derive

What is required is a formula giving the volume,  $V_b$ , in millilitres, of the strong base NaOH of molarity  $C_B$ , added from a burette to  $V_f$  ml of a solution of a weak acid, of initial molarity  $C_{A0}$ , in the titration flask, where  $V_b$  is a function of the pH in the flask. It is assumed that the protonicity of the weak acid in the flask is known as are all the relevant ‘mixed’ acid dissociation constants. The ionic strength of the solution in the flask should be known and assumed effectively constant throughout the titration. Knowing the ionic strength is essential in calculating the ‘mixed’ acid dissociation constants (as  $pK_m$ ’s here) using the **Derive** expression **PKMN** (#12) and also in calculating the appropriate activity coefficient of hydrogen ions  $\gamma_{H^+}$  using the **Derive** function **ACT\_COEF\_H** (#10).  $\gamma_{H^+}$  in **Derive** formulae has been written as  $\gamma H$ .

Throughout the titration, the electroneutrality conditions in the titration flask, for mono-, di- and triprotic acids are given by Equations (14), (15) and (16) respectively.-

$$[Na^+] + [H^+] = [A^-] + [OH^-] \tag{14}$$

$$[Na^+] + [H^+] = [HA^-] + 2[A^{2-}] + [OH^-] \tag{15}$$

$$[Na^+] + [H^+] = [H_2A^-] + 2[HA^{2-}] + 3[A^{3-}] + [OH^-] \tag{16}$$



In Equations (14), (15) and (16)-

$$[\text{Na}^+] = \frac{C_B V_b}{V_f + V_b} \quad (17)$$

$$[\text{H}^+] = \frac{a_{\text{H}^+}}{\gamma_{\text{H}^+}} \quad (18)$$

$$[\text{OH}^-] = \frac{a_{\text{OH}^-}}{\gamma_{\text{OH}^-}} \quad (19)$$

The ‘thermodynamic’ ionic product for water is given by  $K_w = a_{\text{H}^+} a_{\text{OH}^-}$ , so  $[\text{OH}^-] = \frac{K_w}{a_{\text{H}^+} \gamma_{\text{OH}^-}}$ . As

$\gamma_{\text{OH}^-} \approx \gamma_{\text{H}^+}$ , Equation (19) can be written as-

$$[\text{OH}^-] = \frac{K_w}{a_{\text{H}^+} \gamma_{\text{H}^+}} \quad (20)$$

For **monoprotic acids**,  $[\text{A}^-] = C_A \alpha_1$  (Equation 8), where  $C_A$  is the concentration of the weak monoprotic acid in the titration flask. As solution is run in from the burette, the total weak acid concentration is diluted from the original molar concentration  $C_{A0}$ . So during the titration-

$$[\text{A}^-] = \left( \frac{C_{A0} V_f}{V_f + V_b} \right) \alpha_1 \quad (21)$$

Substituting (17), (18), (20) and (21) in (14) and extracting  $V_b$  gives-

$$V_b = \frac{\left( \frac{K_w}{a_{\text{H}^+}} + \gamma_{\text{H}^+} C_{A0} \alpha_1 - a_{\text{H}^+} \right) V_f}{\left( a_{\text{H}^+} + \gamma_{\text{H}^+} C_B - \frac{K_w}{a_{\text{H}^+}} \right)} \quad (22)$$

For **diprotic acids**,  $[\text{HA}^-] = C_A \alpha_1$  and  $[\text{A}^{2-}] = C_A \alpha_2$  (Equations 11) for which the volume of solution added from the burette is-

$$V_b = \frac{\left( \frac{K_w}{a_{\text{H}^+}} + \gamma_{\text{H}^+} C_{A0} (\alpha_1 + 2\alpha_2) - a_{\text{H}^+} \right) V_f}{\left( a_{\text{H}^+} + \gamma_{\text{H}^+} C_B - \frac{K_w}{a_{\text{H}^+}} \right)} \quad (23)$$

The corresponding expression for **triprotic acids**, using the  $\alpha_1$ ,  $\alpha_2$  and  $\alpha_3$  from Equations (13), is-

$$V_b = \frac{\left( \frac{K_w}{a_{\text{H}^+}} + \gamma_{\text{H}^+} C_{A0} (\alpha_1 + 2\alpha_2 + 3\alpha_3) - a_{\text{H}^+} \right) V_f}{\left( a_{\text{H}^+} + \gamma_{\text{H}^+} C_B - \frac{K_w}{a_{\text{H}^+}} \right)} \quad (24)$$



Equations (22), (23) and (24) correspond to the **Derive** Equations **VOL\_BUR\_MP** (#24), **VOL\_BUR\_DP** (#35) and **VOL\_BUR\_TP** (#48) in the utility file. These apply to the titration of mono-, di- and triprotic acids respectively.

### Locating end-points of titrations using Derive's DIF\_DATA

The end point inflections in acid-base titrations may be difficult to read accurately. Sometime a first derivative of the titration curve is used to be able to obtain an accurate value of this end point volume. The functions below illustrate this approach.

**VOL\_BUR\_MP**, **VOL\_BUR\_DP** and **VOL\_BUR\_TP** result in 2 column matrices in which the first column is the volume,  $V_b$ , (ml) of titrant added from the burette and the second column is the pH in the titration flask. Using **Derive**'s **DIF\_DATA** function, the second column is numerically converted to the first differential,  $\frac{\delta(pH)}{\delta V_b}$ .  $V_b$  can be written as  $V_i$ , where  $i$  is the  $i$ th volume value in the first col-

umn of the matrix. The first differential matrices are included in the **Derive** utility file as follows.

**FDMP := DIF\_DATA(VOL\_BUR\_MP(Vf, CA0, CB, pKw, pKm1, γH))**

**FDDP := DIF\_DATA(VOL\_BUR\_DP(Vf, CA0, CB, pKw, pKm1, pKm2, γH))**

**FDTP := DIF\_DATA(VOL\_BUR\_TP(Vf, CA0, CB, pKw, pKm1, pKm2, pKm3, γH))**

**FDMP\_SCALED**(lencol, multdif), **FDDP\_SCALED**(lencol, multdif) and **FDTP\_SCALED**(lencol, multdif) (#51, #55 and #59 in the utility file) enable the first differentials in the second columns, of **FDMP** etc, to be multiplied by a scaling factor, 'multdif', so that they can be plotted at a suitable scale on the titration graphs. The parameter 'lencol' is the column length of the matrix, given by **lencol := DIM(FDMP, COL 2)**, etc.

### Buffer Capacity of the solution during the titration

The buffer capacity, or buffer index, is a measure of the ability of a solution to resist changes in pH due to the addition of strong acids or bases. If  $C_{SB}$  is the concentration of strong base added, and  $C_{SA}$  the concentration of strong acid added, both in equivalents per litre, then the buffer capacity is defined by-

$$\beta = \frac{d(C_{SB} - C_{SA})}{d(pH)} \quad (25)$$

In the solutions titrated here, only strong base is added, so we can write-

$$\beta = \frac{d(C_{SB})}{d(pH)} \quad (26)$$

Here,  $C_{SB} = [\text{Na}^+] = \frac{C_B V_b}{(V_f + V_b)}$ , where  $V_b$  is the volume of NaOH solution added from the burette.

Differentiating  $C_{SB}$  with respect to  $V_b$  gives-

$$\frac{d(C_{SB})}{d(V_b)} = \frac{C_B V_f}{(V_f + V_b)^2} \quad (27)$$

Writing  $V_f$  as  $V_i$ , the volume added values in the first column of the matrices **FDMP**, **FDDP** and **FDTP**, then the buffer capacity can be written as-

$$\beta = \frac{d(C_{SB})}{d(pH_i)} = \frac{d(C_{SB})}{dV_i} \frac{dV_i}{d(pH_i)} = \frac{C_B V_f}{(V_f + V_i)^2} \frac{dV_i}{d(pH_i)}$$

$$\text{So, } \beta = \frac{\left( \frac{C_B V_f}{(V_f + V_i)^2} \right)}{\left( \frac{d(pH)}{dV} \right)_i} \quad (28)$$

The **Derive** function **DIF\_DATA** has already been used to calculate a 2 column matrix in which the first column is  $V_i$  values and the second column,  $\left( \frac{d(pH)}{dV} \right)_i$ . Hence the buffer capacity graph can be

obtained via (28). The  $\beta$  values are orders of magnitude too small to be plotted on the pH titration graph so they are first scaled by a factor 'multbuf'. The buffer capacity curves can be created and scaled using the utility file functions **BUFCAP\_MP**(lencol,multbuf), (#52), **BUFCAP\_DP**(lencol,multbuf) (#56) and **BUFCAP\_TP**(lencol,multbuf) (#60) for mono-, di- and triprotic acid titrations respectively.

## Reference

1. Pankow, J. F., 1991, Aquatic Chemistry Concepts, Lewis Publishers, Chelsea, Michigan, USA, pp 46 – 49.

## (2) Weak Acid Utility File: Creating Speciation and Titration Graphs for Mono-, Di- and Triprotic Acids

File Weak Acid utility file CDK.mth is among the files connected with this DNL and can be downloaded.

## (3) Some Examples using the Utility File functions: Malonic Acid, a weak diprotic acid.

```
#1: LOAD(C:\Program Files\TI Education\Derive 6\Users\\Weak Acid utility file
CDK.mth)
```

It is supposed that we have a 0.002 M solution of malonic acid which also contains the strong electrolyte KCl at a molar concentration of 0.1. Except at extreme low and high pH, the ionic strength, I, will be determined by the KCl present. So-

```
#2: conc := 0.1
```

<b>D-N-L#59</b>	<b>Colin Kennedy: Weak Acid Speciation a. o. with Derive</b>	<b>p17</b>
-----------------	--	------------

```
#3:  num_cat := 1
#4:  cat_charge := 1
#5:  an_charge := -1
#6:  SING_SALT_I(conc, num_cat, cat_charge, an_charge)
#7:                                     0.1
#8:  I := 0.1
```

The 'thermodynamic' dissociation constants for malonic acid at 298 K are-

```
#9:  pKt1 := 2.85
#10: pKt2 := 5.7
```

At 298 K,  $A = 0.51$ , the charge on the fully protonated acid, 'z' is 0, and for the first dissociation step,  $n = 1$ . So the first 'mixed' acid dissociation constant is given by-

```
#11: A := 0.51
#12: PKMN(pKt1, A, I, 0, 1)
#13:                                     2.737670932
```

The second 'mixed' acid dissociation constant is-

```
#14: PKMN(pKt2, A, I, 0, 2)
#15:                                     5.363012797
```

So, to 3 significant figures-

```
#16: pKm1 := 2.74
#17: pKm2 := 5.36
```

## Molar Concentration versus pH Graphs

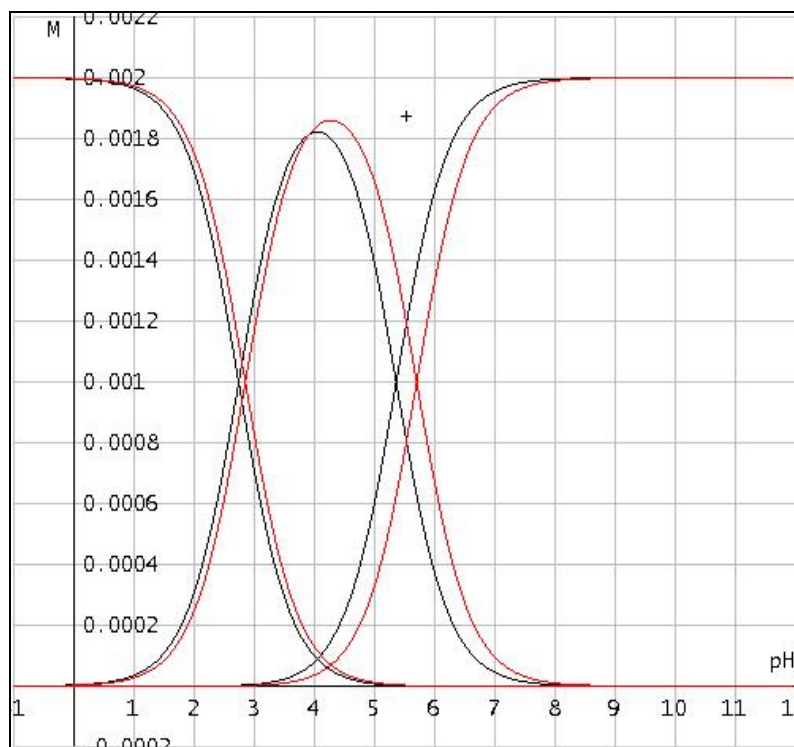
For a 0.002 M solution of the acid the individual species concentrations may now be plotted as a function of pH. (Black curves in the graph)

```
#18: CA := 0.002
#19: H2A(CA, pKm1, pKm2)
#20: HAm(CA, pKm1, pKm2)
#21: Am2(CA, pKm1, pKm2)
```

At very low ionic strength, the  $pK_m$  and  $pK_t$  values will be almost equal, so the speciation curves would then be obtained from-

```
#22: H2A(CA, pKt1, pKt2)
#23: HAm(CA, pKt1, pKt2)
#24: Am2(CA, pKt1, pKt2)
```

**These are the red curves in the graph.** Note that a change in ionic strength can make a significant difference to the species concentrations.



### pH Titration Graphs

Suppose that ( $V_f = 20$ ) ml of ( $CA_0 = 0.002$ ) M malonic acid are titrated against ( $CB = 0.02$ ) M NaOH solution in the burette. All solutions contain 0.1 M KCl, so the ionic strength  $I$ , is still 0.1. The activity coefficient of hydrogen ions,  $\gamma_H$ , at this ionic strength is first required-

#25:  $ACT\_COEF\_H(A, I)$

#26: 0.7720953423

#27:  $\gamma_H := 0.772$

The titration curve can now be calculated as a two column matrix (The black curve on the graph).-

#28:  $V_f := 20$

#29:  $CA_0 := 0.002$

#30:  $CB := 0.02$

#31:  $pK_w := 14$

#32:  $VOL\_BUR\_DP(V_f, CA_0, CB, pK_w, pK_{m1}, pK_{m2}, \gamma_H)$

The first differential matrix of this is calculated (FDDP) and the column length, (lenco1) found.-

#33: FDDP

#34:  $lenco1 := DIM(FDDP\ COL\ 2)$

#35:  $lenco1 := 101$

The vertical height of the first differential plot may need to be scaled to fit it on the same graph. Possibly a separate scaling may be necessary for each endpoint (**red curves**).

```
#36: multdif := 0.1
```

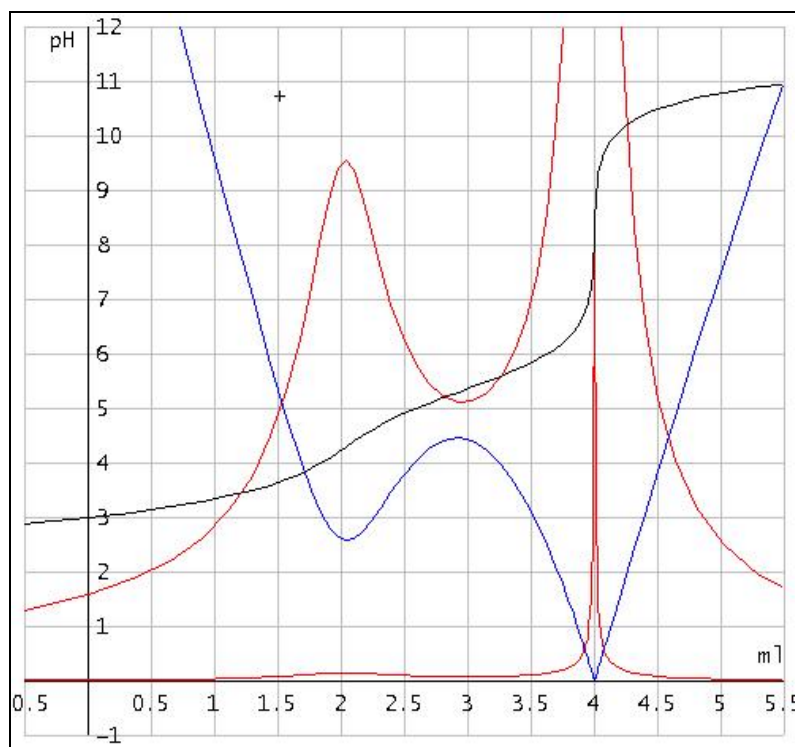
```
#37: FDDP_SCALED(lencol, multdif)
```

Changing multdif to 6 gives the higher red curve, revealing the first less clear endpoint.

The buffer capacity of the solution in the titration flask changes throughout the titration, becoming higher as free hydrogen or hydroxyl ion concentrations increase, and often reaching a peak at or near **half** equivalence point titres (as in commonly used buffer solutions). This has to be scaled by a large factor (multbuf) to make it visible on the graph (**blue curve**).

```
#38: multbuf := 5000
```

```
#39: BUFCAP_DP(lencol, multbuf)
```



pH titrations are often used to find the pK<sub>m</sub>'s and hence the K<sub>m</sub>'s of weak acids by reading off the pH at half equivalence points, here at 1.00 ml and 3.00 ml of titrant added. This method yields apparent values read from the graph of pK<sub>m1</sub> = 3.35 and pK<sub>m2</sub> = 5.38, instead of the actual values of 2.74 and 5.36. So the estimation of the second dissociation constant exponent is reasonable but the first is significantly in error. Discussion with students as to why the method fails helps to encourage (1), better understanding of the solution chemistry involved and (2), a more judicious approach to experimental methods.

## Log Concentration versus pH Graphs

It is often useful to show speciation graphs as plots of  $\log_{10}(\text{molar concentration})$  versus pH. One advantage of this approach is that the full range of hydrogen ion and hydroxyl ion molar concentrations can be shown on the same graph as that of the other species present. The functions to be plotted are-

For  $[\text{H}^+]$  (red line) :-

$$\#40: -\text{pH} - \text{LOG}(\gamma_{\text{H}}, 10)$$

For  $[\text{OH}^-]$  (blue line)

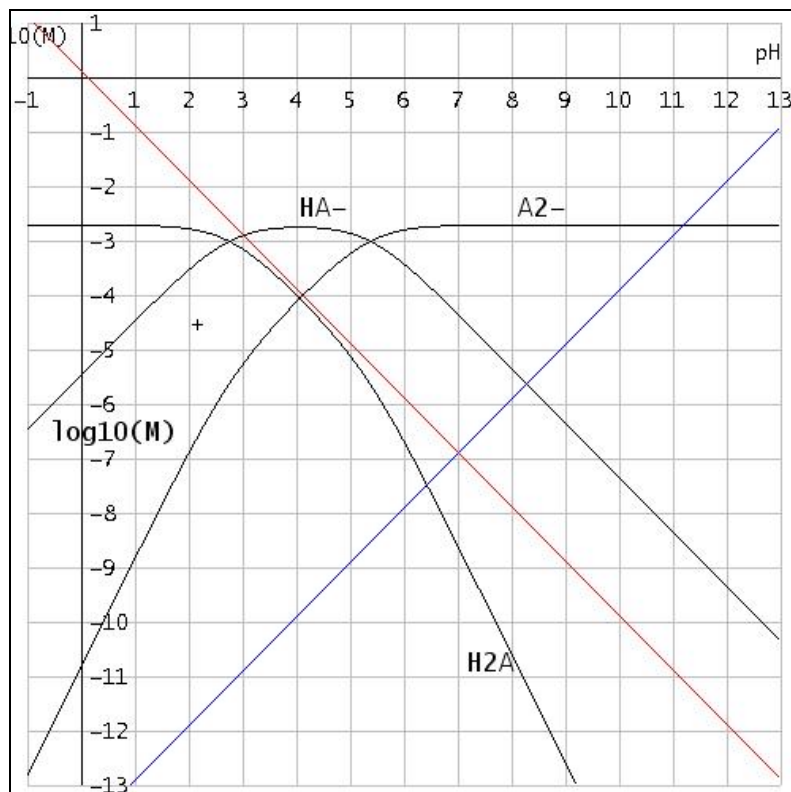
$$\#41: -\text{pKw} + \text{pH} - \text{LOG}(\gamma_{\text{H}}, 10)$$

For the 3 species derived (pun not intended!) from malonic acid (black lines on the graph) ;-

$$\#42: \text{LOG}(\text{H}_2\text{A}(\text{CA}, \text{pK}_{\text{m1}}, \text{pK}_{\text{m2}}), 10)$$

$$\#43: \text{LOG}(\text{HAm}(\text{CA}, \text{pK}_{\text{m1}}, \text{pK}_{\text{m2}}), 10)$$

$$\#44: \text{LOG}(\text{Am}_2(\text{CA}, \text{pK}_{\text{m1}}, \text{pK}_{\text{m2}}), 10)$$



The concentration values below pH 2 and above pH 12 are less accurate as the ionic strength of the solution is no longer largely determined by the 0.1 M KCl in the solution. Nevertheless, relationships between  $[\text{H}^+]$  and  $[\text{OH}^-]$  with the concentrations of the species derived from malonic acid are made apparent from these graphs.

As my main intention is to show that Derive can be a useful tool in teaching some aspects of solution chemistry, I have provided only a few practical examples. However, there are an almost infinite number of ways in which the generation of graphs similar to those above can simulate practical experiments as well as stimulate an understanding of the principles involved.

### Comments on the “Differential Matrix”

The author of the “Weak Acid ...”-contribution mentions a “Differential Matrix”(33). Inspecting his Utility file we find that he applies the DERIVE function DIF\_DATA which is part of **NumericalApproximation.mth**. This is a nice way for numerical differentiation if a function is given by a consecutive sequence of data points. We can read in the Online – Help:

DIF\_DATA(A) approximates to the first derivative of a numerical data matrix A. A must have two columns and at least three rows, with each successive row being a successive [x, y] coordinate pair. The result is an analogous data matrix wherein the second column is the approximate first derivative of y with respect to x at successive values of x. The x values can be unequally spaced. They can even be nonmonotonic, such as samples from a parametrically-defined curve. The formula is based on the derivatives of parabolas that fit three successive points.

It might be useful for readers (and for students) to doublecheck the procedure: We take three consecutive data points and find the parabola passing through the points. The 1<sup>st</sup> derivative in the 2<sup>nd</sup> point is an approximation for the derivative of the unknown function containing all data points. There are two exceptions: For the first point one has to take the derivative of the parabola given by the first three points in the first point – and similar for the last point. Let’s check with DERIVE:

We create the matrix of data points (101 points) and name it m.

```
#86: m := VOL_BUR_DP(20, 0.002, 0.02, 14, pKm1, pKm2, 0.772)
#87: m
```

The quadratic regression line through the first three points serves for two purposes:

$$\begin{bmatrix} -7.674608567 & 2 \\ -6.547594726 & 2.1 \\ -5.483814654 & 2.2 \\ -4.498347975 & 2.3 \end{bmatrix}$$

It gives the slope for the first and for the second point.

```
#89: p1 := FIT([x, a·x2 + b·x + c], [m1, m2, m3])
```

```
#90: p1 := 0.002407500835·x2 + 0.1229700212·x + 2.801945902
```

```
#91:  $\frac{d}{dx} p1 = 0.004815001670 \cdot x + 0.1229700211$ 
```

Points #2, #3 and #4 define a parabola and its first derivative in the middle point is the slope of the unknown function in point #3.

```
#92: SUBST( $\frac{d}{dx} p1, x, m_{1,1}$ ) = 0.08601676813
```

```
#93: SUBST( $\frac{d}{dx} p1, x, m_{2,1}$ ) = 0.09144334165
```

DIF\_DATA uses another algorithm but comparing the first lines of DIF\_DATA(m) which is FDDP in the contribution shows the correctness of the procedure.

```
#94: p3 := FIT([x, a·x2 + b·x + c], [m2, m3, m4])
```

```
#95: SUBST( $\frac{d}{dx} p3, x, m_{3,1}$ ) = 0.09788232195
```

```
#96: DIF_DATA(m)
```

See the TI-realisation on page 39.

$$\begin{bmatrix} -7.674608567 & 0.08601676815 \\ -6.547594726 & 0.09144334167 \\ -5.483814654 & 0.09788232195 \end{bmatrix}$$

# Wie wär's mit einem Pringle?

## A Pringle is a Mathematical Potato Chip!

Benno Grabinger, Neustadt/Weinstraße, [www.bennograbinger.de](http://www.bennograbinger.de)



Wie kann die Form eines Pringle mathematisch beschrieben werden?

Wo entstehen solche Fragen?

Eine Kursfahrt nach Florenz und Siena war der Ausgangspunkt:

*How can we describe the form of a Pringle mathematically?*

*Where are such questions coming from?*

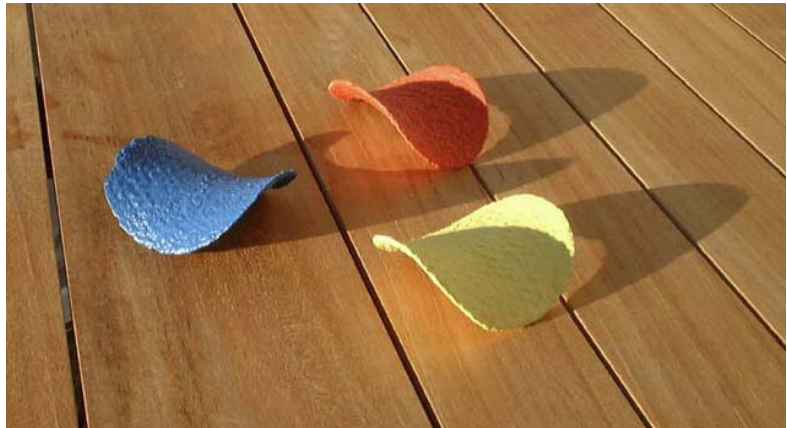
*A class excursion to Florence and Siena was the initial point.*





Gipsmodelle (hergestellt von Sebastian Schulz, Leibniz-Gymnasium Neustadt, 2004) erleichtern das Arbeiten.

Gypsum models (produced by Sebastian Schulz) support imagination.



Offenbar sind zwei Symmetrieebenen vorhanden, welche das Pringle längs der gestrichelten Linien schneiden.

Der Einfachheit halber soll einmal angenommen werden, dass die eingezeichneten Schnitte der Symmetrieebenen mit dem Pringle Parabeln sind. Wählt man den Schnittpunkt der eingezeichneten Linien als Ursprung eines geeigneten Koordinatensystems, so können die Gleichungen der Parabeln ermittelt werden, wenn man das Pringle ausmisst.

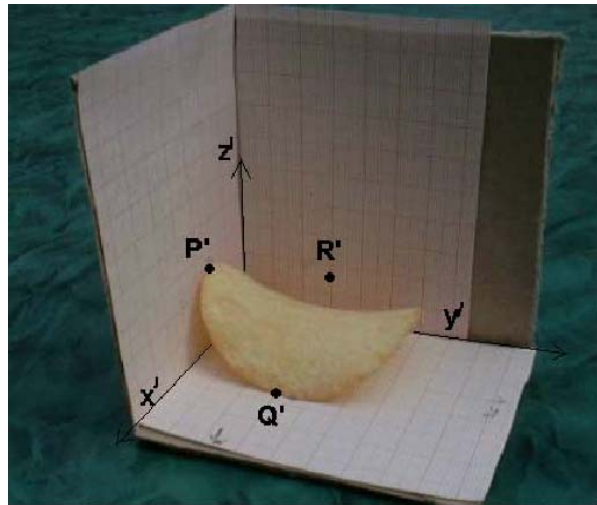
*Obviously there are two planes of symmetry, which intersect the Pringle along the dashed lines.*

*To make things easier we will assume that these dashed intersection lines are parabolas. Choosing their intersection point as origin of an appropriate system of coordinates we should be able to find out the equations of the parabolas by measuring the Pringle.*

*(For the following I'll try to translate Benno's paper into English to save space. Josef. As I wrote in the Letter of the Editor, the full German version can be downloaded.)*

We take the bearings of the center of symmetry of the pringle and we obtain point  $R'$  (0/3/1.5) as projection of the center on the  $y$ - $z$ -plane (related to the  $x'y'z'$  system of coordinates in the vertex of space).

Points  $P'$  and  $Q'$  are  $P'(2.3/0/2)$  and  $Q'(4.5/3/0)$ .



As we want to work with a system of coordinates  $xyz$  with its center in the center of symmetry of the pringle and axes parallel to  $x'y'z'$  we have to consider the translation  $\bar{v} = \begin{pmatrix} 2.3 \\ 3 \\ 1.5 \end{pmatrix}$ . Then

points  $P$  and  $Q$  have coordinates  $P(0/-3/0.5)$  and  $Q(2.2/0/-1.5)$ .

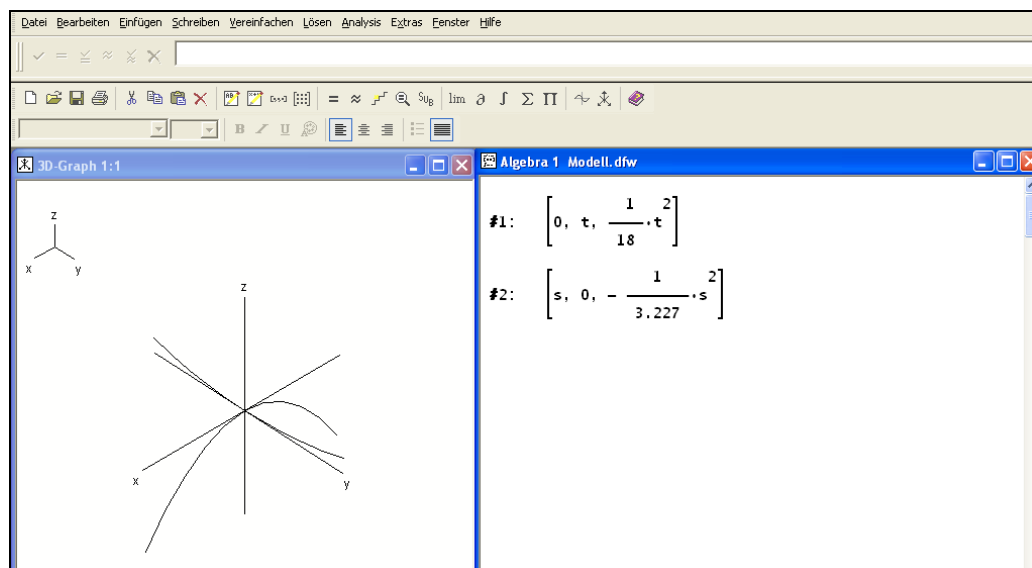
The parabola passing point  $P$  lying in  $yz$ -plane has the general form  $z = a y^2$  and from  $0.5 = a * 9$  follows that  $a = 1/18$ .

So finally we obtain  $z = \frac{1}{18} y^2$  or in parameter form  $\bar{x} = \begin{pmatrix} 0 \\ t \\ \frac{1}{18} t^2 \end{pmatrix}$ .

Very similar we find the parabola with vertex  $Q$  lying in the  $xz$ -plane,  $z = b x^2$ :  
 $-1.5 = b * 2.2^2$  which leads to  $b = -75/242$ .

$$z = -\frac{75}{242} x^2 \rightarrow z = -\frac{1}{3.227} x^2 \rightarrow \bar{x} = \begin{pmatrix} s \\ 0 \\ -\frac{1}{3.227} s^2 \end{pmatrix}.$$

Plot of the parameter representations.  
[Derive file modell.dfw](#)

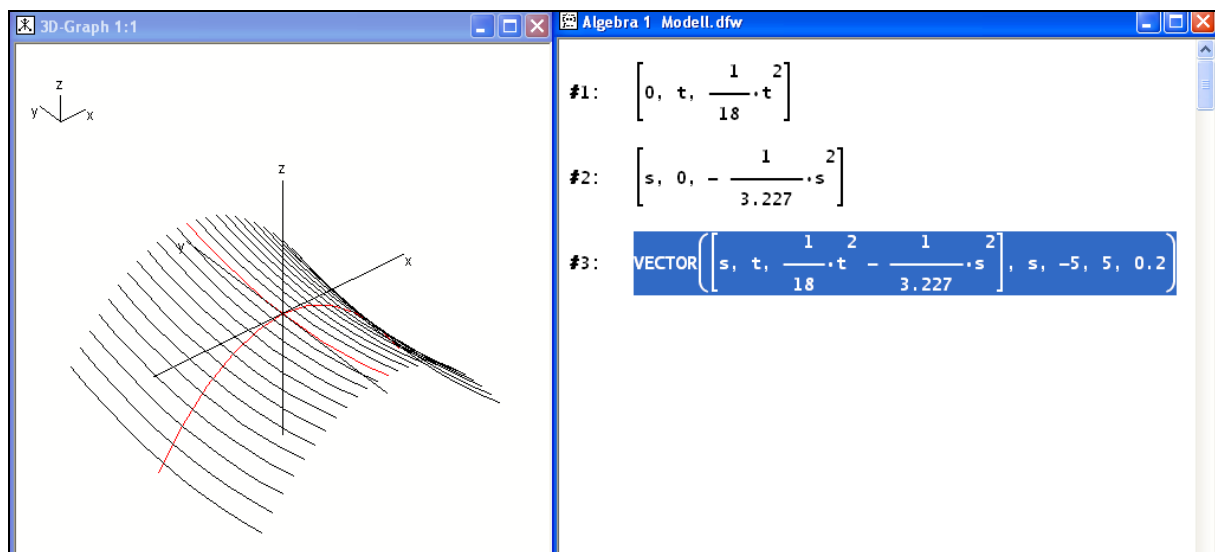


Which is the next step?

Inspecting the intersection curves of the pringle with planes parallel to the  $yz$ -plane one get the impression that these curves might be also parabolas and in the easiest case these parabolas could be the same as the parabola in the  $yz$ -plane. In order to reach this goal in the model one has to shift this first parabola in such a way that its vertex glides along the parabola lying in the  $xz$ -plane.

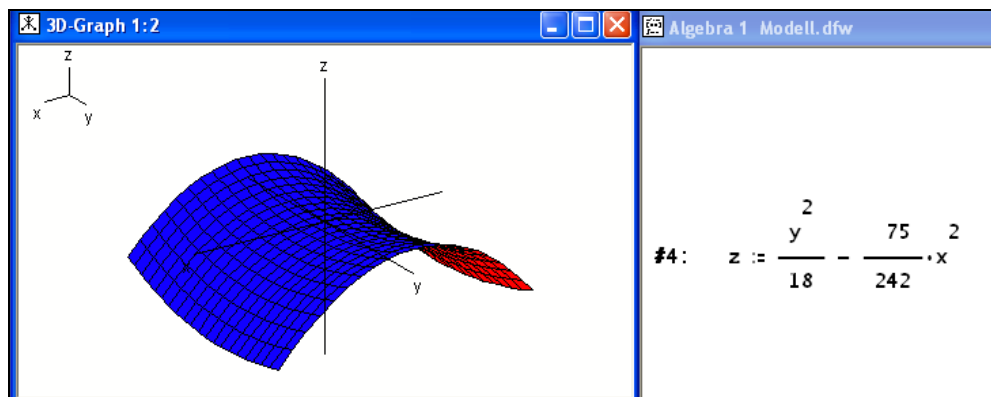
This is achieved by adding a translation vector to the parameter form of the parabola being shifted. We find the translation vector by reasoning how the origin is moved. As the origin shall glide along the second parabola we have:

$$\vec{v} = \begin{pmatrix} s \\ 0 \\ -\frac{s^2}{3.227} \end{pmatrix} \text{ and further } \vec{x} = \begin{pmatrix} 0 \\ t \\ \frac{t^2}{18} \end{pmatrix} + \begin{pmatrix} s \\ 0 \\ -\frac{s^2}{3.227} \end{pmatrix} = \begin{pmatrix} s \\ t \\ \frac{t^2}{18} - \frac{s^2}{3.227} \end{pmatrix} \quad (*)$$



You can see a family of parabolas which are translated with an increment of  $s = 0.2$ .

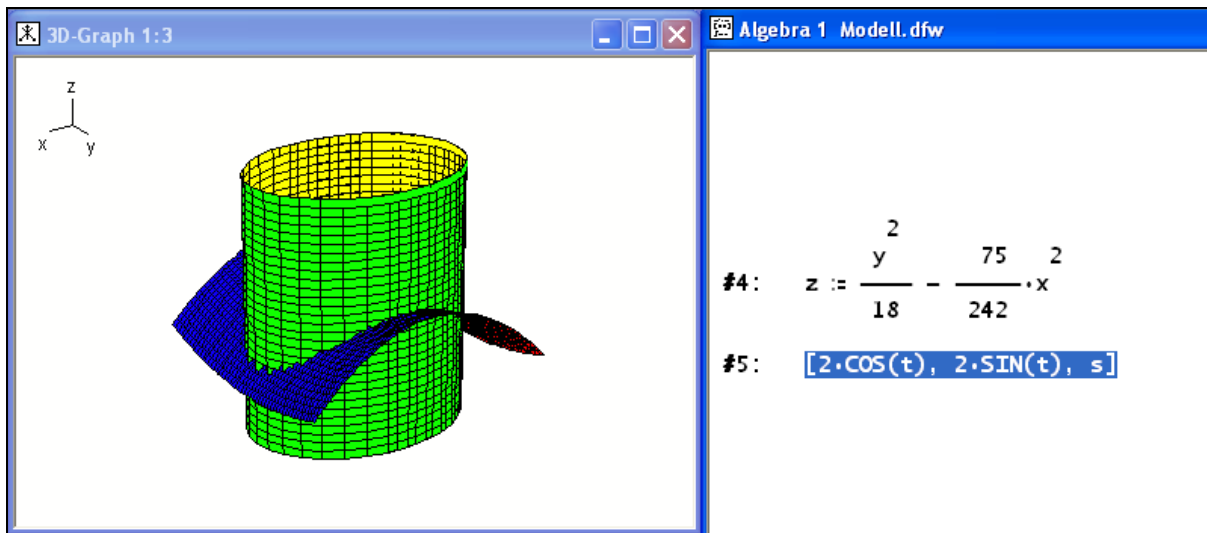
The parameter form (\*) leads immediately to  $z = \frac{y^2}{18} - \frac{75x^2}{242}$ .



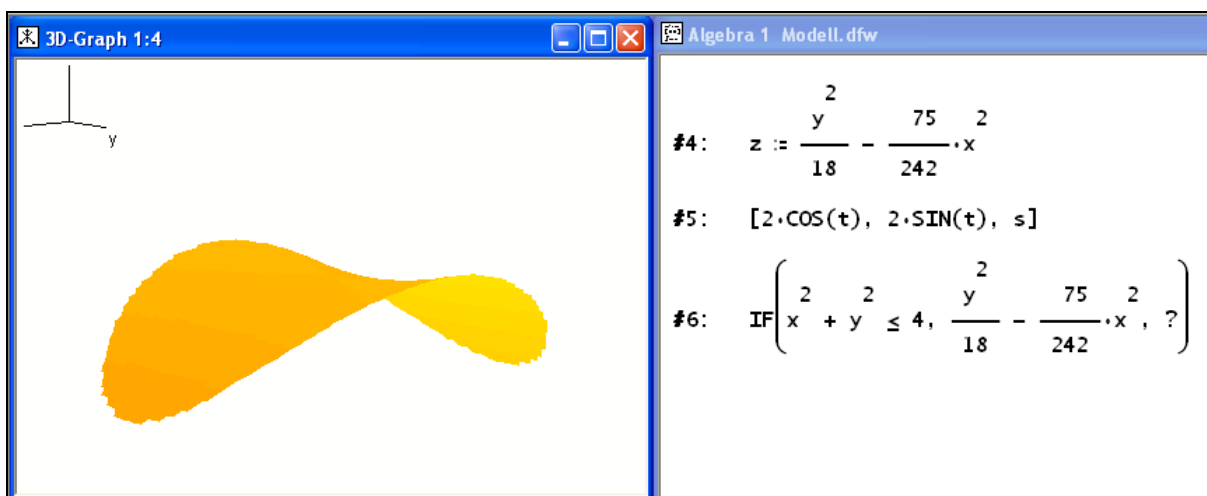
This surface is very similar to the pringle but the boundary lines are missing:

Pringles must fit into a can in the shape of a cylinder, so we have the idea to intersect the

surface  $z = \frac{y^2}{18} - \frac{75x^2}{242}$  with a cylinder which has the z-axis as its axis.



Observing the intersection of cylinder and surface we obtain the Pringle:

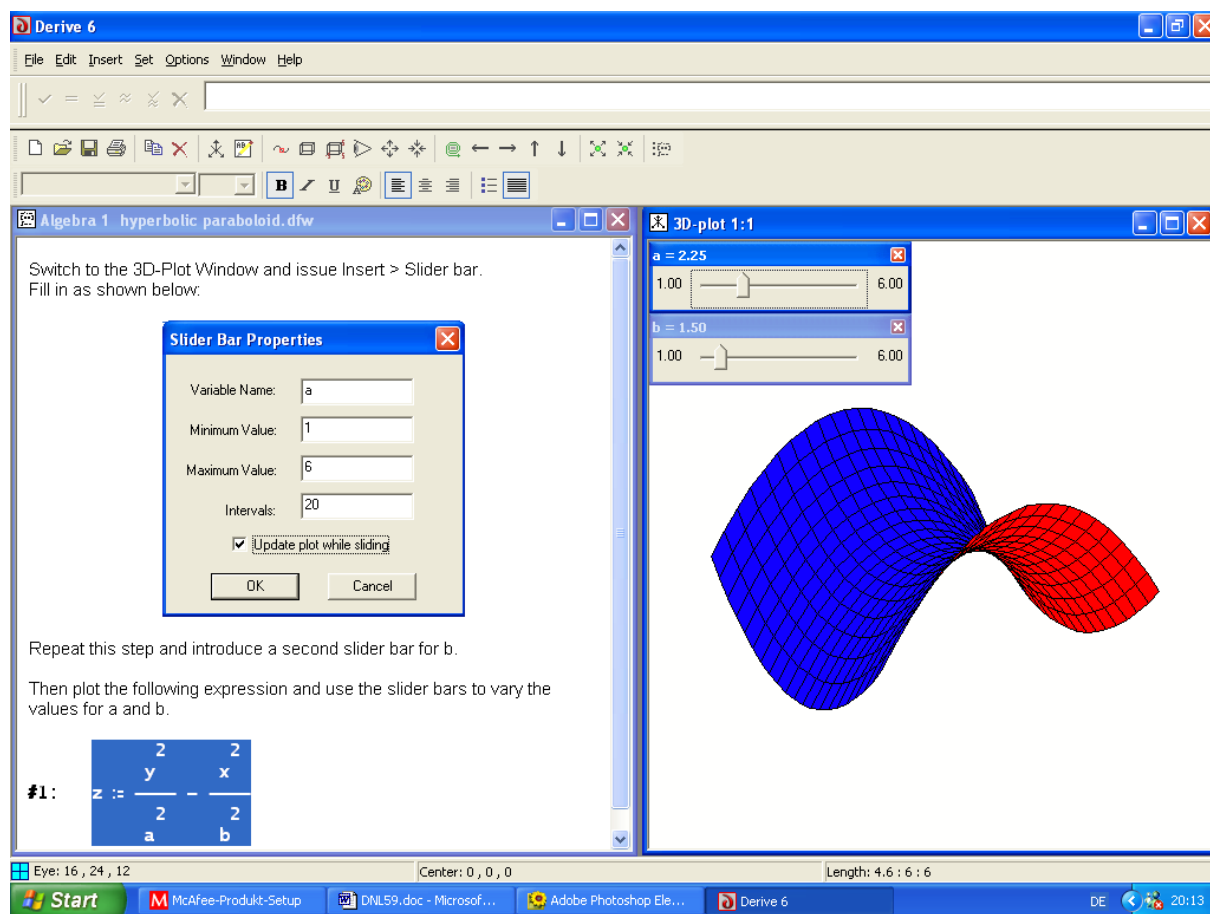


(Expression #6 shows how to "produce" the Pringle. For this purpose we use the rectangular form of the cylinder. We don't show the mesh lines and choose an appropriate color in order to plot a realistic "naturally coloured" picture of the Pringle.)

The equation which is the base of the Pringle  $z = \frac{y^2}{18} - \frac{75x^2}{242} = \frac{y^2}{18} - \frac{x^2}{3.227}$  is a special case of the so called **hyperbolic paraboloid**  $z = \frac{y^2}{a^2} - \frac{x^2}{b^2}$ .

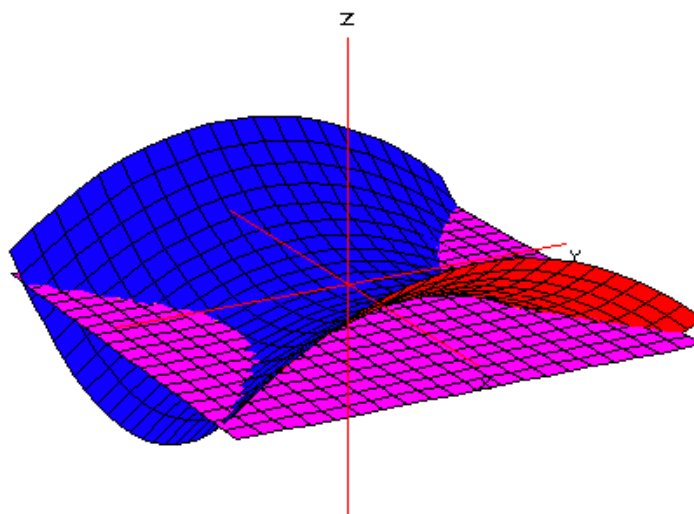
One can study the influence of constants  $a$  and  $b$  for the form of this surface by introducing slider bars.

[Derive file hyperbolic paraboloid.dfw](#)



## The Hyperbolic Paraboloid

The word "paraboloid" is self explanatory because of the gliding parabolas which form the surface. The attached adjective "hyperbolic" is based on the form of the intersection curves of the surface with horizontal planes.



It seems to be very likely that the intersection curve is a hyperbola. Taking the plane  $z = k$  then the intersection curve is given by  $k = \frac{y^2}{a^2} - \frac{x^2}{b^2}$ .

## Examples

1. For  $k = -1$  (picture above) we find:

$$-1 = \frac{y^2}{a^2} - \frac{x^2}{b^2} \rightarrow 1 = \frac{x^2}{b^2} - \frac{y^2}{a^2}$$

And this is the equation of a hyperbola in the  $xy$ -system of coordinates.

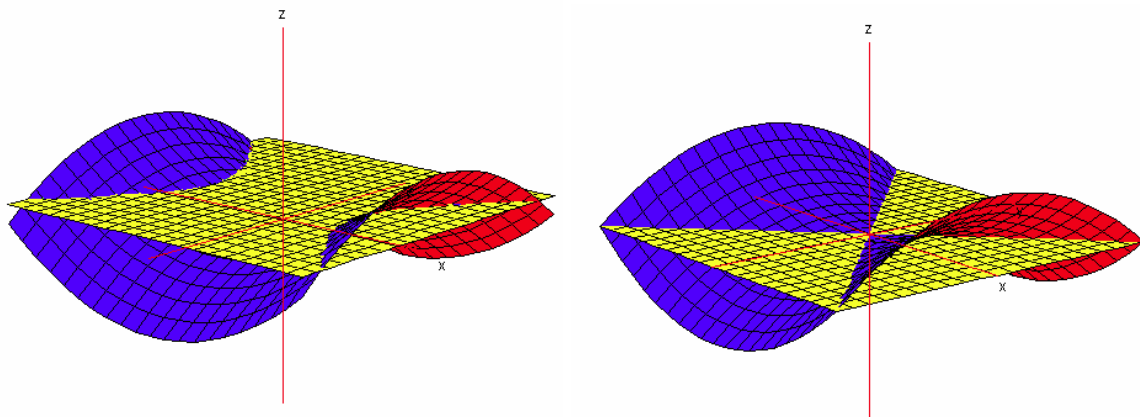
2. For  $k = 1$  (left graph in the next picture ) we find:

$$1 = \frac{y^2}{a^2} - \frac{x^2}{b^2} \text{ which is again a hyperbola in the } yx\text{-system of coordinates.}$$

3. For  $k = 0$  (right graph in the next picture) we have a special case:

$$0 = \frac{y}{a} \pm \frac{x}{b},$$

i.e. the intersection curve falls into two lines.



These two lines are called the *main vertex generating lines* (*Hauptscheitelerzeugende*) of the hyperbolic paraboloid. Their intersection point lies on the axis of the surface and is called *vertex*.

The following properties of this surface can immediately be derived from its equation

$$z = \frac{y^2}{a^2} - \frac{x^2}{b^2} :$$

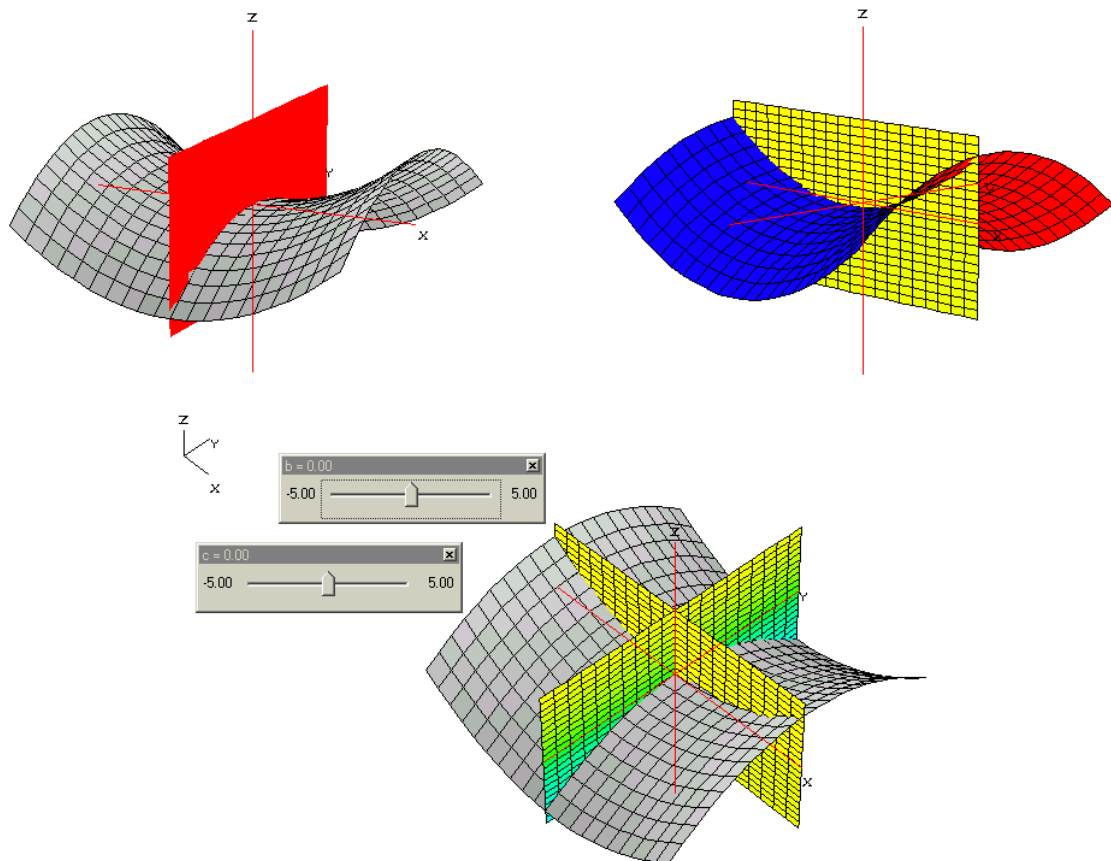
There is exact one point of the surface above each point of the top-view-plane ( $xy$ -plane).

The  $yz$ -plane  $x = 0$  is symmetry plane of the surface, because variable  $x$  occurs only in quadratic form, i.e. if point  $P(x/y/z)$  is point of the surface then point  $P^*(-x/y/z)$  is also a point of the surface.

The  $xz$ -plane  $y = 0$  is symmetry plane of the surface, too, because variable  $y$  occurs only in quadratic form, i.e. if point  $Q(x/y/z)$  is point of the surface then point  $Q^*(x/-y/z)$  is also a point of the surface.

The intersection line of the two planes of symmetry – in our presentation the  $z$ -axis – is an axis of symmetry of the surface. It is called *Axis* of the hyperbolic paraboloid.

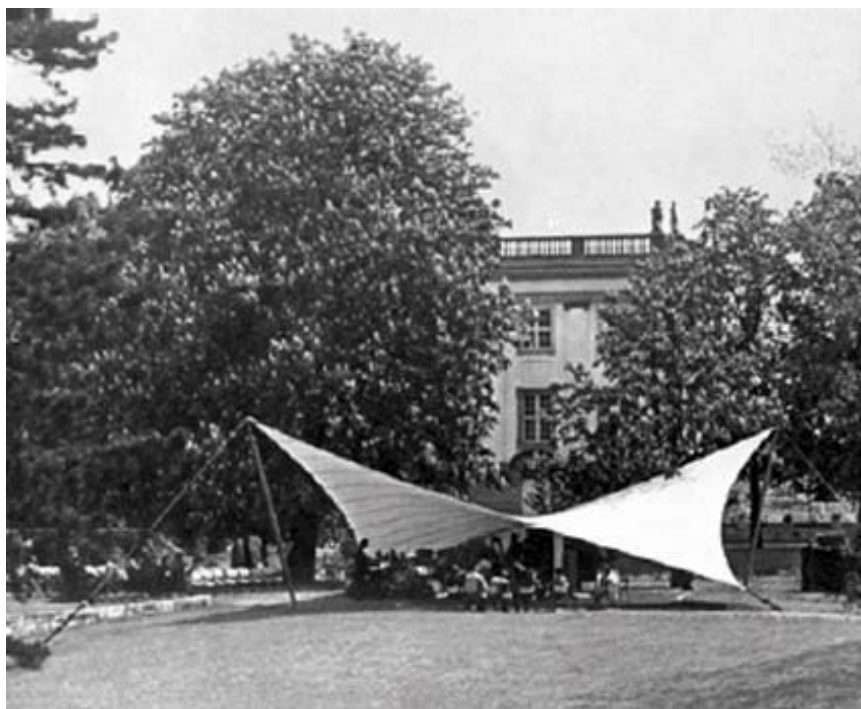




[Derive file intersection.dfw](#)

There are a lot of applications of this surface in architecture.

See some examples:



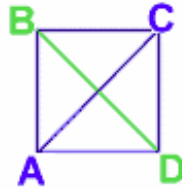
Saddle Surface: Membrane-roof for the Bundesgartenschau in Cologne, 1957

On the website <http://www.stellenboschwriters.com/> one can find:

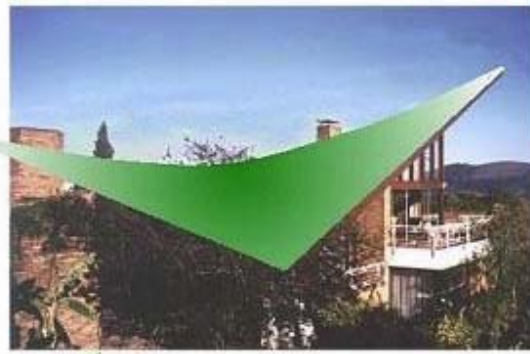
### What is a Hyperbolic Paraboloid?

Very easy, it is the mathematical name for the form of a saddle.

Take a piece of textile.



Bend vertices A and C upwards and fix them. Then bend vertices B and D downwards. If your textile is large enough you can use it as a sun sail as we did on board of our yacht. This sun sail shows the same form. It is called HP-surface (HP-Schale).

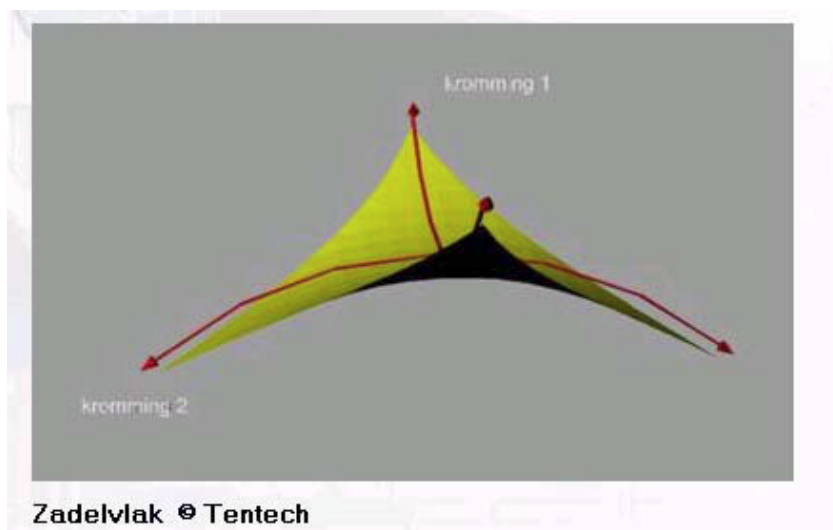


Another example:

<http://www.bingham.nl/en/enfotozadel.html>

### Saddle Surface

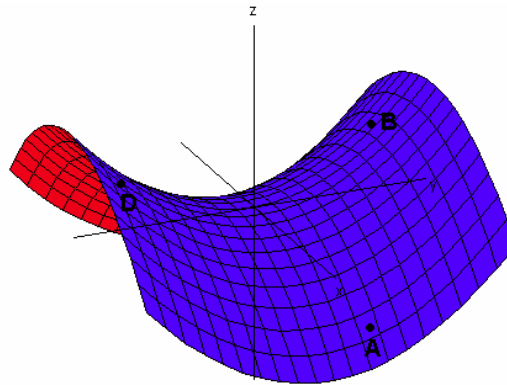
The saddle surface consists of two maximum- and two minimum points with a membrane tightened between them.





The given examples deliver another approach to the hyperbolic paraboloid. We will show that the form produced by the described method of hanging up a textile gives a paraboloid.

Take the paraboloid  $z = \frac{y^2}{k^2} - \frac{x^2}{k^2}$ . Four points A, B, C and D on the surface are selected in such a way that A and C have  $z$ -coordinate  $-h$  and  $y$ -coordinate 0. Points B and D are above the  $xy$ -plane with  $z$ -coordinate  $h$  and  $x$ -coordinate 0.  
(See the graph below with hidden point C)



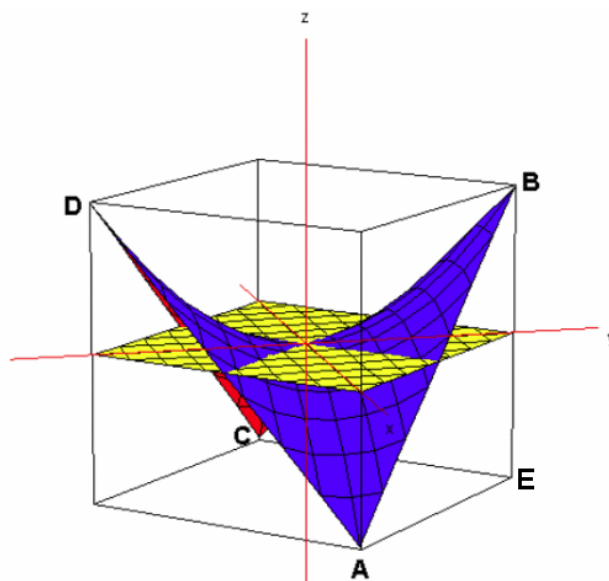
B is defined by  $B(0/y/h)$ .

For calculating  $y$  we substitute the coordinates of B in  $z = \frac{y^2}{k^2} - \frac{x^2}{k^2}$ :

$$y^2 = h k^2 \text{ or } |y| = k \sqrt{h}. \text{ So } B(0/k\sqrt{h}/h) \text{ and } D(0/-k\sqrt{h}/h).$$

In a similar way we obtain  $A(k\sqrt{h}/0/-h)$  and  $C(-k\sqrt{h}/0/-h)$ .

If we intersect the paraboloid with a cube defined by its vertices A, B, C and D then the resulting form is very similar to the suspended membrane which we mentioned earlier.



How to obtain this intersection figure?

Point E has coordinates  $E(0/k\sqrt{h}/-h)$ .

The plane defined by A, B and E:  $\vec{r} = \vec{r}_A + \overline{AB} + \overline{AE}$ .

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} k\sqrt{h} \\ 0 \\ -h \end{pmatrix} + m \begin{pmatrix} -k\sqrt{h} \\ k\sqrt{h} \\ 2h \end{pmatrix} + n \begin{pmatrix} -k\sqrt{h} \\ k\sqrt{h} \\ 0 \end{pmatrix}$$

This leads to  $x + y = k\sqrt{h}$ . In the same way one can calculate the equations of the remaining planes of the cube. We collect the conditions for the inner part of the cube:

$$|x + y| \leq k\sqrt{h} \text{ and } |x - y| \leq k\sqrt{h} \text{ and } |z| \leq h$$

[Derive file suspended.dfw](#)

The last figure leads to the conjecture that lines  $g(A,B)$  and  $g(C,D)$  are lying on the surface – that they are part of the surface. This conjecture is checked by calculation:

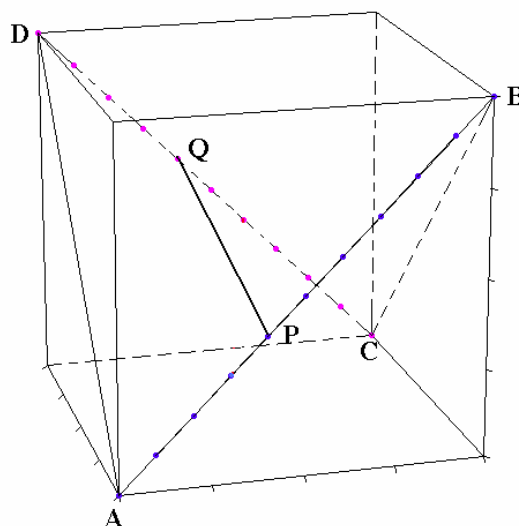
$$g(A,B): \vec{r} = \begin{pmatrix} k\sqrt{h} \\ 0 \\ -h \end{pmatrix} + m \begin{pmatrix} -k\sqrt{h} \\ k\sqrt{h} \\ 2h \end{pmatrix}. \text{ Substituting the coordinates into } z = \frac{y^2}{k^2} - \frac{x^2}{k^2} \text{ results in:}$$

$$\frac{y^2}{k^2} - \frac{x^2}{k^2} = \frac{m^2 k^2 h}{k^2} - \frac{(k\sqrt{h} - mk\sqrt{h})^2}{k^2} = m^2 h - (\sqrt{h} - m\sqrt{h})^2 = -h + 2mh = z, \text{ i.e. all points of line } g(A,B) \text{ are elements of the hyperbolic paraboloid.}$$

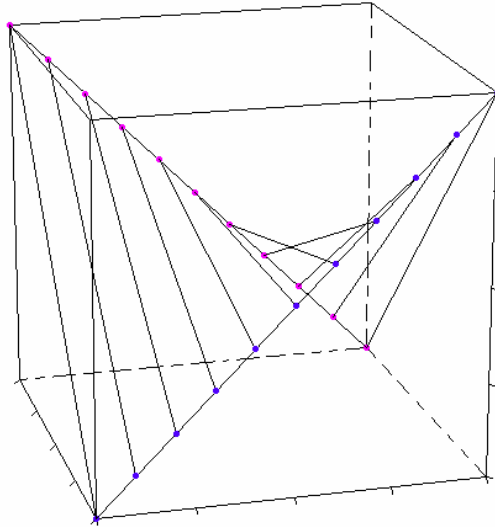
Similar calculation shows that all points of line  $g(C,D)$  are also elements of the surface. The result of the next consideration is, that there are more straight lines which are elements of the surface.

1. Set up the equations for segments AB and DC.
2. Subdivide AB and CD into 10 segments of equal length.
3. Set up the equation for segment PQ, with P being the  $i^{\text{th}}$  division point of AB and Q being the  $i^{\text{th}}$  division point of DC.

Then line  $g(PQ)$  is part of the surface.



Perform this procedure for all division points.



The proof:

$$\text{Segment AB: } \vec{x} = \begin{pmatrix} k\sqrt{h} \\ 0 \\ -h \end{pmatrix} + s \begin{pmatrix} -k\sqrt{h} \\ k\sqrt{h} \\ 2h \end{pmatrix} \text{ with } s \in [0;1], \text{ division points } s = 0.1, 0.2, \dots, 0.9$$

$$\text{Segment DC: } \vec{x} = \begin{pmatrix} 0 \\ -k\sqrt{h} \\ h \end{pmatrix} + s \begin{pmatrix} -k\sqrt{h} \\ k\sqrt{h} \\ -2h \end{pmatrix} \text{ with } s \in [0;1], \text{ division points } s = 0.1, 0.2, \dots, 0.9$$

Straight line connecting division points  $P_s$  and  $Q_s$  (with  $s$  fixed and  $t$  variable)

$$\begin{aligned} \vec{x} &= \begin{pmatrix} k\sqrt{h} \\ 0 \\ -h \end{pmatrix} + s \begin{pmatrix} -k\sqrt{h} \\ k\sqrt{h} \\ 2h \end{pmatrix} + t \left[ \begin{pmatrix} 0 \\ -k\sqrt{h} \\ h \end{pmatrix} + s \begin{pmatrix} -k\sqrt{h} \\ k\sqrt{h} \\ -2h \end{pmatrix} - \begin{pmatrix} k\sqrt{h} \\ 0 \\ -h \end{pmatrix} - s \begin{pmatrix} -k\sqrt{h} \\ k\sqrt{h} \\ 2h \end{pmatrix} \right] = \\ &= \begin{pmatrix} k\sqrt{h} \\ 0 \\ -h \end{pmatrix} + s \begin{pmatrix} -k\sqrt{h} \\ k\sqrt{h} \\ 2h \end{pmatrix} + t \begin{pmatrix} -k\sqrt{h} \\ -k\sqrt{h} \\ 2h \end{pmatrix} + s \cdot t \begin{pmatrix} 0 \\ 0 \\ -4h \end{pmatrix} \quad (**) \end{aligned}$$

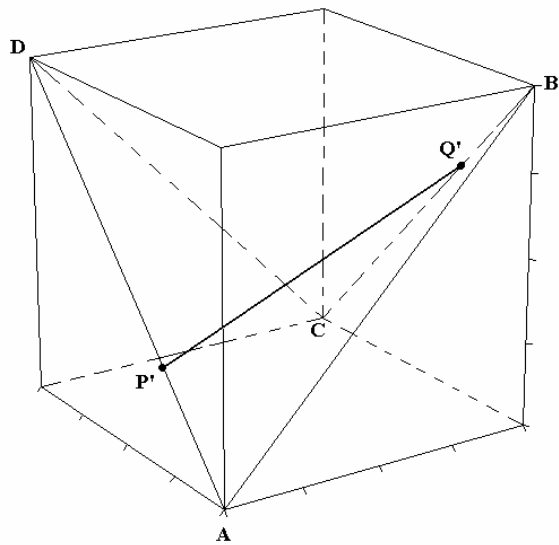
This set of points is completely part of the HP-surface because

$$\begin{aligned} x &= k\sqrt{h} - sk\sqrt{h} - tk\sqrt{h} \\ y &= sk\sqrt{h} - tk\sqrt{h} \\ z &= -h + 2sh + 2th - 4st h \end{aligned}$$

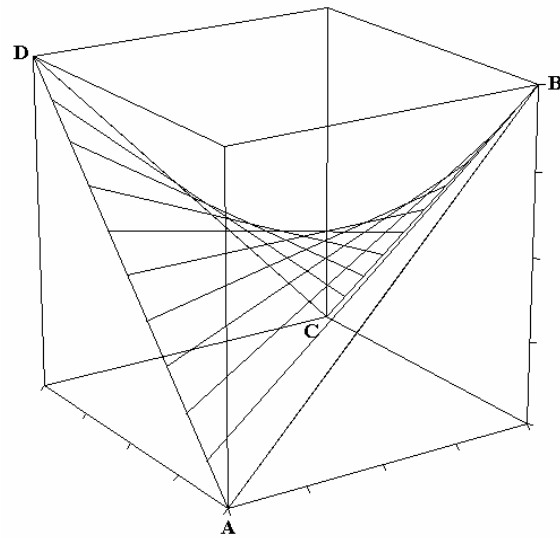
$$\begin{aligned} \frac{y^2}{k^2} - \frac{x^2}{k^2} &= \frac{k^2(s\sqrt{h} - t\sqrt{h})^2}{k^2} - \frac{k^2(\sqrt{h} - s\sqrt{h} - t\sqrt{h})^2}{k^2} = \\ &= h(s^2 - 2st + t^2 - ((1-s) - t)^2) = -h - 2sh + 2th + st(-4h) = z \end{aligned}$$

What happens if keeping  $t$  fixed and  $s$  variable with  $s \in [0;1]$ ?

Answer: for one fixed  $t$

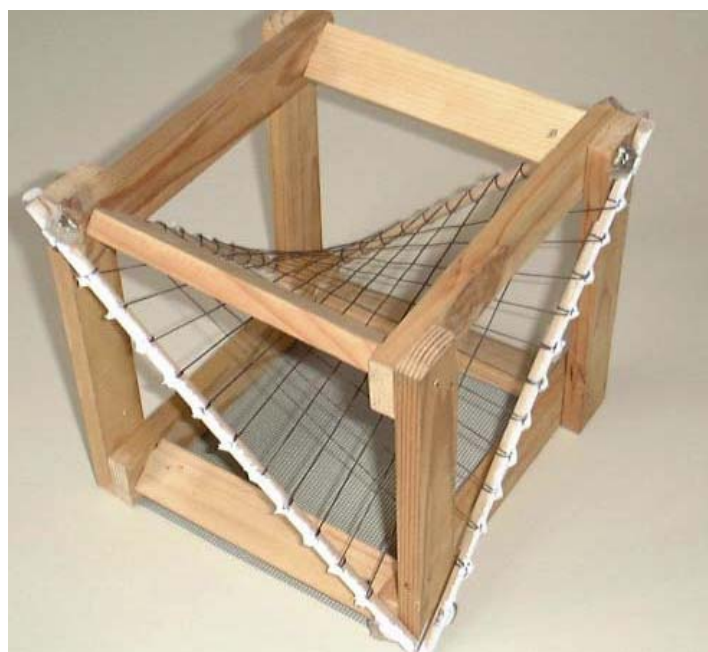
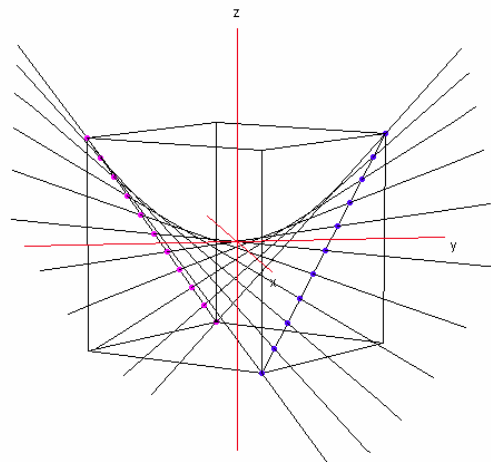


for various values for  $t$ :



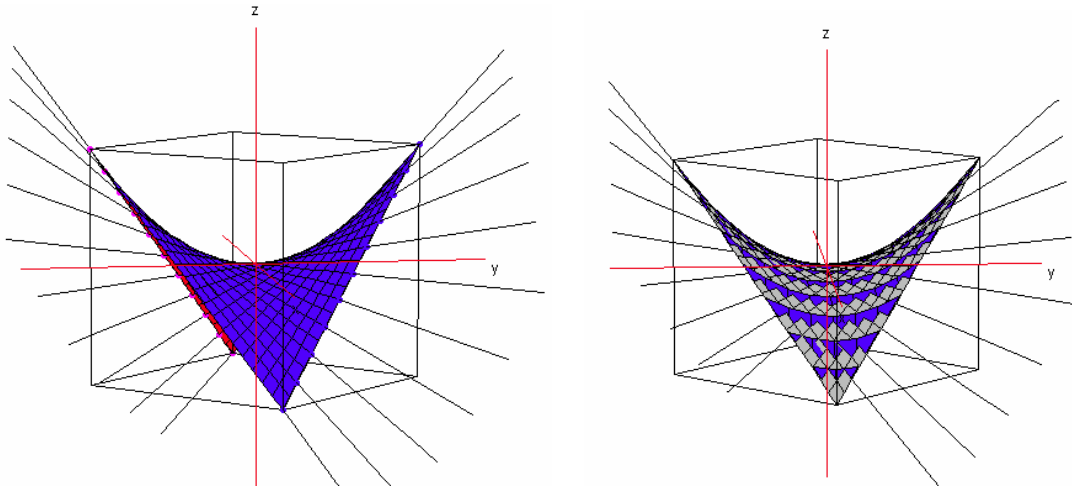
Varying  $s$  and  $t$  in the interval  $[0;1]$  results in two families of lines lying completely in the surface.

Better than a graph is a model, of course.  
(This model was built by Dirk Barthel, a student of the Leibniz-Gymnasium Neustadt, 2004)



Changing the discrete steps for  $s$  and  $t$  to a continuous transition we find the graph as follows in the left graph below.

It cannot be distinguished between this surface and  $z = \frac{y^2}{k^2} - \frac{x^2}{k^2}$  (\*\*\*) which can be shown by adding its graph to the existing one (right graph below).



[Derive file membran1.dfw](#)

The plots are illustrating that both sets of points defined by (\*\*) and (\*\*\*) are identical.

The proof by calculation is very easy:

We showed above that each point of (\*\*) is also point of (\*\*\*). We will add that each point of (\*\*\*) is also point of (\*\*).

Let  $P\left(x/y/\frac{y^2}{k^2} - \frac{x^2}{k^2}\right)$  an arbitrary point of (\*\*\*). Parameter form of points of (\*\*) is:

$$\#1: \quad x_c := x = k \cdot \sqrt{h} - s \cdot k \cdot \sqrt{h} - t \cdot k \cdot \sqrt{h}$$

$$\#2: \quad y_c := y = s \cdot k \cdot \sqrt{h} - t \cdot k \cdot \sqrt{h}$$

$$\#3: \quad z_c := z = -h + 2 \cdot s \cdot h + 2 \cdot t \cdot h - 4 \cdot s \cdot t \cdot h$$

$$\#4: \quad y_c - x_c = (y - x = \sqrt{h} \cdot k \cdot (2 \cdot s - 1))$$

$$\#5: \quad \text{SOLVE}(y - x = \sqrt{h} \cdot k \cdot (2 \cdot s - 1), s) = \left[ s = -\frac{x - y - \sqrt{h} \cdot k}{2 \cdot \sqrt{h} \cdot k} \right]$$

$$\#6: \quad y_c + x_c = (x + y = \sqrt{h} \cdot k \cdot (1 - 2 \cdot t))$$

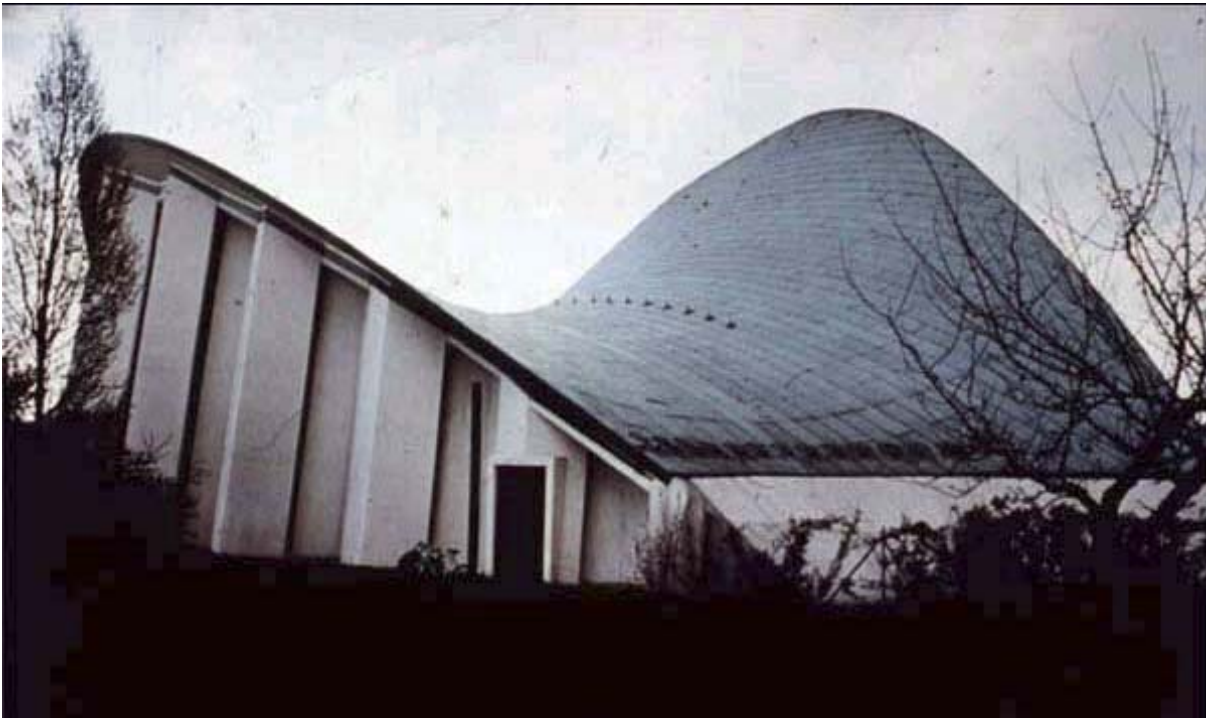
$$\#7: \quad \text{SOLVE}(x + y = \sqrt{h} \cdot k \cdot (1 - 2 \cdot t), t) = \left[ t = -\frac{x + y - \sqrt{h} \cdot k}{2 \cdot \sqrt{h} \cdot k} \right]$$

$$\#8: \quad \text{EXPAND}\left(\text{SUBST}\left(z_c, [s, t], \left[-\frac{x - y - \sqrt{h} \cdot k}{2 \cdot \sqrt{h} \cdot k}, -\frac{x + y - \sqrt{h} \cdot k}{2 \cdot \sqrt{h} \cdot k}\right]\right)\right)$$

$$\#9: \quad z = \frac{y^2}{k^2} - \frac{x^2}{k^2}$$

[Derive file membran2.dfw](#)

The following pictures show some buildings with hyperbolic paraboloids:



Church St. Suitbert, Essen-Überruhr, Architect Josef Lembroek

<http://www.tragwerkstatt.de/ipp/projekte/kat-09/09-03.htm>

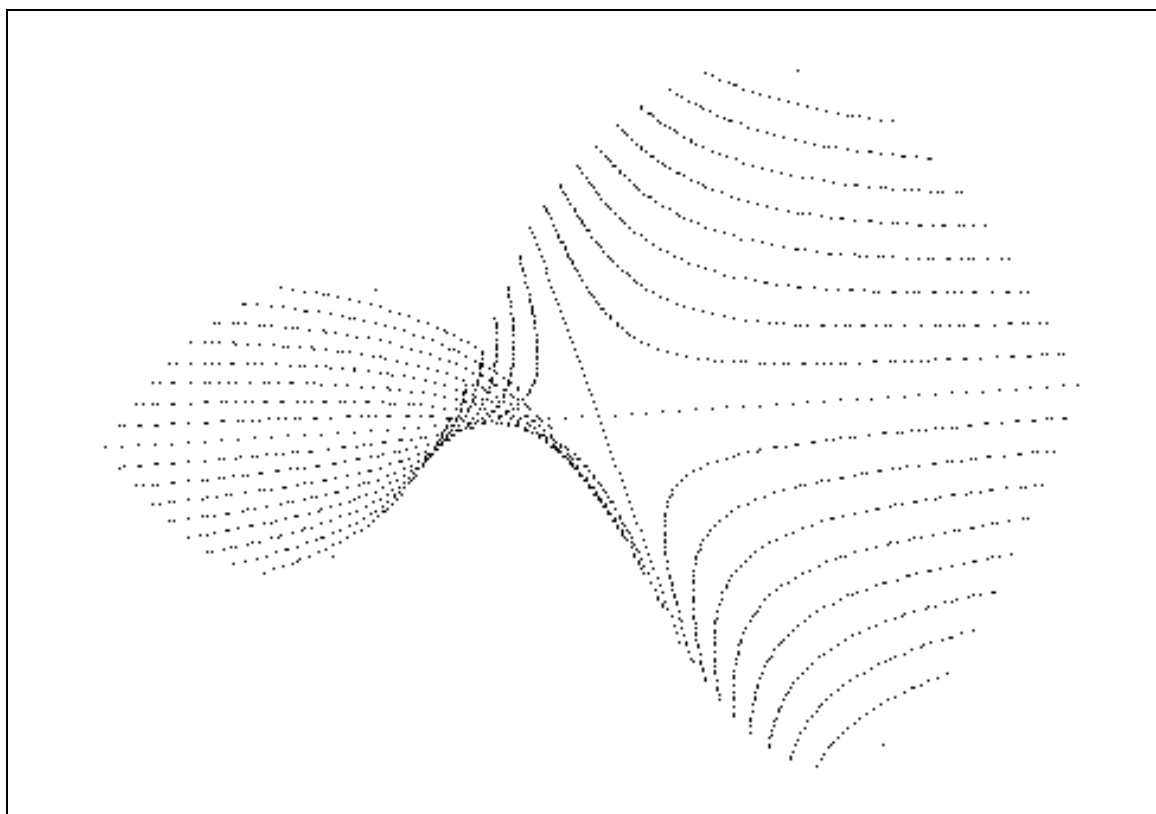


So called "Saddledome" in Calgary, Canada

The next pictures show the restaurant LOS MANANTIALES in Xchochimilco, a suburb of Mexico-City. It was built by Felix Candela in 1958. The roof of this building is composed of hyperbolic paraboloids. (The person on the picture is the architect of this contribution, Benno Grabinger).







A contour plot of the hyperbolic paraboloid

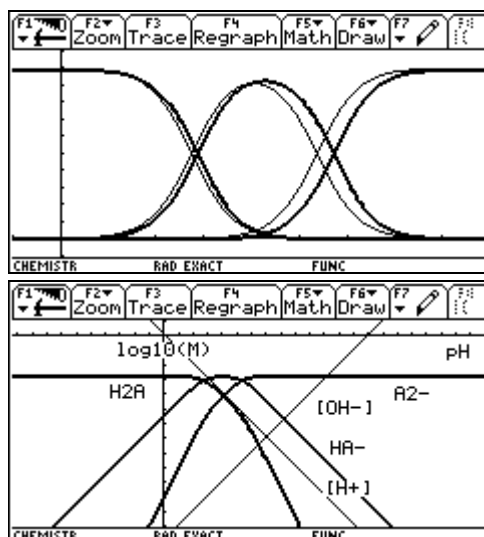
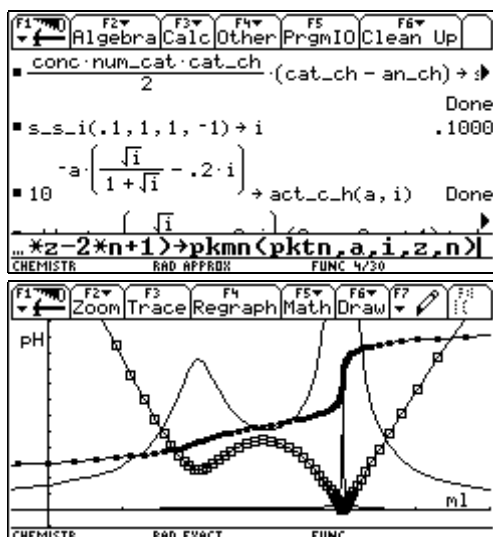


## Text file for CAS-TIs: Weak Acid Speciation and Titrations

```

C:conc*num_cat*cat_ch/2*(cat_ch-an_ch)→s_s_i(conc,num_cat,cat_ch,an_ch)
C:s_s_i(.1,1,1,-1)→i
C:10^(-a*(√(i)/(1+√(i))-.2*i))→act_c_h(a,i)
C:pktn+a*(√(i)/(1+√(i))-.2*i)*(2*z-2*n+1)→pkmn(pktn,a,i,z,n)
:
C:2.85→pkt1:5.7→pkt2
C:.51→a
C:pkmn(pkt1,a,i,0,1)→pkm1
C:pkmn(pkt2,a,i,0,2)→pkm2
:
C:1/(10^(ph-pkm1_)+1)→mpa0(pkm1_)
C:1/(10^(pkm1_-ph)+1)→mpa1(pkm1_)
C:ca*mpa0(pkm1_)→ha(ca,pkm1_)
C:ca*mpa1(pkm1_)→am(ca,pkm1_)

```



```

difdata(m)
Prgm
Local i,d,l_1,l_2
dim(m)[1]→d
newMat(dim(m)[1],2)→m_
{m[1,1],m[2,1],m[3,1]}→l_1
{m[1,2],m[2,2],m[3,2]}→l_2
QuadReg 1_1,l_2
d(regeq(x),x)|x=m[1,1]→m_[1,2]
m[1,1]→m_[1,1]
For i,2,d-1
  {m[i-1,1],m[i,1],m[i+1,1]}→l_1
  {m[i-1,2],m[i,2],m[i+1,2]}→l_2

```

```

QuadReg 1_1,l_2
d(regeq(x),x)|x=m[i,1]→m_[i,2]
m[i,1]→m_[i,1]
EndFor
{m[d-2,1],m[d-1,1],m[d,1]}→l_1
{m[d-2,2],m[d-1,2],m[d,2]}→l_2
QuadReg 1_1,l_2
d(regeq(x),x)|x=m[d,1]→m_[d,2]
m[d,1]→m_[d,1]
m_→fddp
Disp "fddp calculated"
EndPrgm

```

You can find all functions together with the executable script – the first lines of the script are on top of the page - among the downloadable files.

From **Subject:** The Benefit of Multivalued Constants  
to **Subject:** Infinite Sets and Intervals

A report about an exciting email exchange between CAS-experts

**From:** [Albert D. Rich](#)

**To:** [David Jeffrey](#)

**Sent:** Thursday, 21 July, 2005 16:06

Hello David,

I whole-heartedly agree with your conclusion in your and Norman's paper entitled "Multivalued functions in computer algebra" that such functions lead to all kinds of problems and should be avoided.

However, if properly implemented, I have been driven to the conclusion that multivalued constants are indispensable to good computer algebra and mathematics. For years now Derive has had multivalued constants for "plus-or-minus one" and for "an arbitrary point on the unit circle in the complex plane". For example, the equation

$$\text{ABS}(z) = 1$$

simplifies to  $z = \text{unit\_circle}$ . The attached MultivaluedConstants.dfw file illustrates that the "plus-or-minus one" constant is essential for doing even basic algebra. Check it out...

Over the last several months I have incorporated the new multivalued constants #Z and #R into Derive. They represent an arbitrary element of the integers and of the reals, respectively. The algebra required to simplify expressions involving these constants affect the entire system. Some rules are obvious, such as if  $n$  is an integer,

$$\#Z + n \rightarrow \#Z$$

and the not so obvious

$$\#Z "+ -" 1/2 \rightarrow \#Z + 1/2$$

and

$$\#Z "+ -" 1/4 \rightarrow \#Z/2 + 1/4$$

Of course, the real benefit of these new constant is the ability to concisely and completely show all the solutions of equation with infinitely many solutions. For example, solving

$$\text{SIN}(x) = 0$$

for  $x$  used to give

$$x = -\pi \vee x = \pi \vee x = 0$$

whereas, Derive will now give

$$x = \pi * \#Z$$

The following are even more dramatic examples of the benefits of #Z:

1.  $\text{SOLVE}(e^x + e^{-x} = 0, x)$  now yields

$$x = i * (\pi * \#Z + \pi/2)$$

instead of

<b>D-N-L#59</b>	<b>From Multivalued Constants to Infinite Sets, a Report</b>	<b>p41</b>
-----------------	--	------------

$$x = -\frac{5 \cdot \pi \cdot i}{2} \vee x = \frac{5 \cdot \pi \cdot i}{2} \vee x = -\frac{3 \cdot \pi \cdot i}{2} \vee x = \frac{3 \cdot \pi \cdot i}{2} \vee x = -\frac{\pi \cdot i}{2} \vee x = \frac{\pi \cdot i}{2}$$

2. SOLVE  $\left( e^{x^2} = e^x, x \right)$  now yields

$$x = 1/2 - \sqrt{(1 + 8 \cdot \pi \cdot i \cdot \#Z)/2} \text{ OR } x = \sqrt{(1 + 8 \cdot \pi \cdot i \cdot \#Z)/2} + 1/2$$

instead of the mysterious

$$\begin{aligned} x = & -\frac{\sqrt{2 \cdot i \cdot \sqrt{(64 \cdot \pi^2 + 1) + 1} - \sqrt{2}}}{4} - \frac{\sqrt{2 \cdot i \cdot \sqrt{(64 \cdot \pi^2 + 1) - 1}}}{4} \vee x = -\frac{\sqrt{2 \cdot i \cdot \sqrt{(64 \cdot \pi^2 + 1) + 1} - \sqrt{2}}}{4} + \\ & \frac{\sqrt{2 \cdot i \cdot \sqrt{(64 \cdot \pi^2 + 1) - 1}}}{4} \vee x = \frac{\sqrt{2 \cdot i \cdot \sqrt{(64 \cdot \pi^2 + 1) + 1} + \sqrt{2}}}{4} - \frac{\sqrt{2 \cdot i \cdot \sqrt{(64 \cdot \pi^2 + 1) - 1}}}{4} \vee x = \\ & \frac{\sqrt{2 \cdot i \cdot \sqrt{(64 \cdot \pi^2 + 1) + 1} + \sqrt{2}}}{4} + \frac{\sqrt{2 \cdot i \cdot \sqrt{(64 \cdot \pi^2 + 1) - 1}}}{4} \vee x = 1 \vee x = 0 \end{aligned}$$

3. SOLVE  $\left( 27^{x^2 + 1} = 243, x \right)$  now yields

$$x = -\sqrt{(2/3 + 2 \cdot \pi \cdot i \cdot \#Z/(3 \cdot \text{LN}(3)))} \text{ OR } x = \sqrt{(2/3 + 2 \cdot \pi \cdot i \cdot \#Z/(3 \cdot \text{LN}(3)))}$$

instead of

$$\begin{aligned} x = & -\frac{\sqrt{6 \cdot i \cdot \sqrt{(\sqrt{\text{LN}(3)^2 + \pi^2} - \pi) + \sqrt{(\text{LN}(3)^2 + \pi^2) + \pi}}}}{6 \cdot \sqrt{\text{LN}(3)}} + \frac{\sqrt{6 \cdot i \cdot \sqrt{(\sqrt{\text{LN}(3)^2 + \pi^2} + \pi) - \sqrt{(\text{LN}(3)^2 + \pi^2) - \pi}}}}{6 \cdot \sqrt{\text{LN}(3)}} \vee x = - \\ & \frac{\sqrt{6 \cdot i \cdot \sqrt{(\sqrt{\text{LN}(3)^2 + \pi^2} - \pi) + \sqrt{(\text{LN}(3)^2 + \pi^2) + \pi}}}}{6 \cdot \sqrt{\text{LN}(3)}} + \frac{\sqrt{6 \cdot i \cdot \sqrt{(\sqrt{\text{LN}(3)^2 + \pi^2} - \pi) - \sqrt{(\text{LN}(3)^2 + \pi^2) + \pi}}}}{6 \cdot \sqrt{\text{LN}(3)}} \vee x = \\ & \frac{\sqrt{6 \cdot i \cdot \sqrt{(\sqrt{\text{LN}(3)^2 + \pi^2} - \pi) + \sqrt{(\text{LN}(3)^2 + \pi^2) + \pi}}}}{6 \cdot \sqrt{\text{LN}(3)}} + \frac{\sqrt{6 \cdot i \cdot \sqrt{(\sqrt{\text{LN}(3)^2 + \pi^2} + \pi) - \sqrt{(\text{LN}(3)^2 + \pi^2) - \pi}}}}{6 \cdot \sqrt{\text{LN}(3)}} \vee x = \\ & \frac{\sqrt{6 \cdot i \cdot \sqrt{(\sqrt{\text{LN}(3)^2 + \pi^2} - \pi) + \sqrt{(\text{LN}(3)^2 + \pi^2) + \pi}}}}{6 \cdot \sqrt{\text{LN}(3)}} + \frac{\sqrt{6 \cdot i \cdot \sqrt{(\sqrt{\text{LN}(3)^2 + \pi^2} + \pi) - \sqrt{(\text{LN}(3)^2 + \pi^2) + \pi}}}}{6 \cdot \sqrt{\text{LN}(3)}} \vee x = -\frac{\sqrt{6}}{3} \vee x = \\ & = \frac{\sqrt{6}}{3} \end{aligned}$$

Now I am considering the ramifications of implementing multivalued constants for an arbitrary element of the nth roots of unity. It will be entered as #Un where n is a positive integer. This will make the solution of equations of the form  $x^n = c$  be the single, concise equation

$$x = \#Un \cdot c^{(1/n)}$$

instead of n messy ones. Note that as an added bonus, #U1 can replace the existing multivalued constant "plus-or-minus one" (of course, it will still be displayed as "+-1").

Food for thought. Look forward to your comments

Aloha,

Albert

p42	From Multivalued Constants to Infinite Sets, a Report	D-N-L#59
-----	---	----------

**From:** [Albert D. Rich](#)

**To:** [David Jeffrey](#)

**Sent:** Thursday, 21 July, 2005 23:25

Oops. Instead of #U1, below I should have said #U2 (i.e. the square-roots of unity) is equivalent to the "plus-or-minus one" multivalued constant. #U1 of course equals 1.

Aloha

**From:** [David Jeffrey](#)

**To:** [Albert D. Rich](#)

**Sent:** Friday, 22 July, 2005 02:09

Hi Albert,

At first glance, I do not see any conflict between my paper and your discoveries. The fact is that some mathematical questions do have multiple solutions. You now have the apparatus to show that to the user. This is a very good improvement to Derive.

David (from hot and polluted China)

**From:** [David Jeffrey](#)

**To:** [Albert D. Rich](#)

**Sent:** Monday, 25 July, 2005 15:52

Hi Albert,

Your  $\pm$  example is interesting. In fact I have used that idea before. I rewrite  $-b \pm \sqrt{b^2 - 4ac}$  as  $-b \pm b \sqrt{(1 - 4ac/b^2)}$  where the  $b^2$  can be pulled out exactly because of the  $\pm$ .

Some questions: if someone wants to plot the solutions, or evaluate them numerically, then you will have to split the symbol into two expressions. How do you think to do this? At present Derive is happy to plot  $y = \sqrt{9 - x^2}$  OR  $y = -\sqrt{9 - x^2}$ , so you could either transform into that format or do it behind the scenes.

In many problems, one solves an equation and then chooses the meaningful branch. It would be good to have a simple way of substituting a definite sign for the  $\pm$  in particular cases.

David

**From:** [Albert D. Rich](#)

**To:** [David Jeffrey](#)

**Cc:** [David Stoutemyer](#)

**Sent:** Monday, 22 August, 2005 20:31

Hi David,

Thanks for your comments.

1. Yes, I need to address the problem of plotting expressions containing multivalued constants in the next release of Derive.
2. After highlighting the  $\pm$  subexpression, use the Simplify > Subexpression substitution command (shortcut Ctrl-T) to replace it with plus or minus of the subexpression.

Any comments about the real meat of my original email (i.e the benefits of #R, #Z, and possibly #Un)?

Aloha,

Albert

**From:** [David Jeffrey](#)

**To:** [Albert D. Rich](#)

**Sent:** Wednesday, 31 August, 2005 10:03

Hi Albert,

I looked again at your letter, and it seems to me there are two separate thoughts here.

1. An equation will sometimes have multiple solutions, and these are best expressed using your functions #R, #Z. Examples are evaluating an integral (solving the simplest differential equation) and solving a trig equation.
2. After that first step of including them in equation solutions, an equally important step is working with them. This is really the new part. It would be interesting to see what other systems do with #R+1, which of course should simplify to #R. So point 1 is I think not controversial, but point 2 is a real challenge I think. So keep working at it!

David

P.S. I may be able to drop in and visit either in December or in March.  
More later.

**From:** [Stoutemyer, David](#)

**To:** [Albert D. Rich](#)

**Sent:** Wednesday, 07 September, 2005 11:17

Hi Albert,

I too have come to the conclusion that we need them, but I have some uncertainties that I thought perhaps we could resolve by brainstorming:

- a) Do you regard, for example  $+1$  as the set  $\{-1, +1\}$  or as one arbitrary element of that set? If it is a set, then what about when the "-" is written over the "+" ( $-1$ ) to mean "consistently take either the top or the bottom choice throughout an expression such as  $(a \pm b)(c \pm d)$ ."
- b) How do we more generally handle correlated multi-valued constants? By this I mean when the same  $\pm$  subexpression occurs several places in an expression. In such instances the "element-of" interpretation yields tighter results than the "set" interpretation.
- c) With the set interpretation, an expression such as  $a \pm b$  means that we are adding a scalar "a" to a set "b", unless we regard "a" as shorthand for  $\{a\}$ , in which case this notational abbreviation might lead to apparent paradoxes.
- d) How does this kind of addition and subtraction come along with union and set differences? I suspect that we shift our interpretation according to context, but I worry that being informal and inconsistent about this will lead to difficulties.
- e) Regrettably, we never made Derive intervals first class citizens, displayable and accessible to the user. Do intervals denote sets or an arbitrary element of a set? How do we combine them with multi-valued constants? How do we combine them with Boolean expressions such as  $0 < x \leq 5$ ?
- f) A single interval is sometimes insufficient for computing limits and representing the solutions of inequalities. The most general multi-valued constant is a "multi-interval" such as  $x = \{[-3,4], 7, [9,\text{inf}]\}$  or else  $x \text{ element\_of } \{[-3,4], 7, [9,\text{inf}]\}$ . We lose information if we generously degrade that multi-interval into  $(-3,\text{inf})$ .
- g) With an interval, how do we indicate that we know an endpoint is tight, versus indicating we know that an endpoint is a bound that isn't tight, versus indicating we don't know whether or not an endpoint is tight? This could matter, for example, in computing limits.

-- aloha, david

**From:** [Albert D. Rich](#)

**To:** [David Jeffrey](#) ; [David Stoutemyer](#)

**Cc:** [Michel Beaudin](#) ; [Bernhard Kutzler](#)

**Sent:** Sunday, 11 September, 2005 23:57

Hello David^2,

Thanks for your interest in the problem of multivalued expressions. We all seem to agree on the importance of properly using and simplifying such expressions.

When David Stoutemyer was here this Friday, he emphasized the importance of finding a notation that would be natural to users. The "+-" operator is the only widely accepted way to display multivalues. Unfortunately, it is woefully inadequate for even some simple multivalued expressions. For example, an arbitrary element of the set {2, 3} in terms of "+-" is

$$(5^{+-}1)/2$$

Representing an arbitrary element of three element sets using "+-" is even less intuitive.

David also pointed out that most people interpret script Z and R as the set of integers and reals, respectively; *not* as an arbitrary element of those sets.

To address both these problems, I propose the following: Make the lower case Greek letter epsilon a unary operator that reads as "an arbitrary element of". Note that the data type of the epsilon operator is the same type as the elements of the set. For example,

$$\text{epsilon}\{2, 3\}$$

is a scalar (i.e. an arbitrary element of the set {2, 3}) that can be included in an expression like any other scalar. Note that epsilon of a single element set is that element.

Since "+-" is a compact and accepted operator, internal expressions of the form

$$\text{epsilon}\{z, -z\}$$

will be displayed as "+-z".

Also I now propose that script Z and R denote the set of integers and reals, respectively; instead of an arbitrary element of those sets like I proposed before. Z and R are entered on the author line as #Z and #R, respectively. Then to get an arbitrary element of those sets, enter

$$\text{epsilon}(\#Z) \quad \text{and} \quad \text{epsilon}(\#R)$$

respectively. For example,

$$\text{SIN}(z) = 0$$

will simplify to

$$z = \pi * \text{epsilon}(\#Z)$$

which reads as "z equals pi times an arbitrary element of the integers".

Traditionally epsilon is a binary operator requiring a variable as its left operand. However, making it a unary operator preserves the utility of the operator, while eliminating the need to introduce a new dummy variable in an expression. Note that the :epsilon binary operator will continue to be used for domain declarations of variables.

So in addition to finite sets (e.g. {a, b, c}), Derive now has the infinite sets #Z and #R. That still leaves open the question how to denote *and* represent arbitrary intervals (i.e. infinite sets of contiguous points).

The obvious notation for denoting open, closed, open-closed, and closed-open intervals would be  $(a, b)$ ,  $[a, b]$ ,  $(a, b]$ , and  $[a, b)$ , respectively. Unfortunately, since square brackets and parentheses are already heavily over-loaded delimiters, using them to denote intervals could be confusing, if not ambiguous. For example, the notation for the closed interval  $[a, b]$  is identical to that of the two-element vector  $[a, b]$ . Also  $F$  times the open interval  $(a, b)$  looks like the function call  $F(a, b)$ .

Then there is the problem of how to represent intervals internally:

An arbitrary element of the interval  $[-1, 1]$  could be represented as  $\text{SIN}(\text{epsilon}(\#R))$ , and the interval itself could be represented by  $\{\text{SIN}(\text{epsilon}(\#R))\}$ . An arbitrary element of the interval  $[0, 1)$  could be represented by  $\text{MOD}(\text{epsilon}(\#R))$ , and the interval itself could be represented by  $\{\text{MOD}(\text{epsilon}(\#R))\}$ . An arbitrary element of the interval  $[0, \text{inf})$  could be represented by  $\text{ABS}(\text{epsilon}(\#R))$ , and the interval itself could be represented by  $\{\text{ABS}(\text{epsilon}(\#R))\}$ .

Then any arbitrary closed interval  $[a, b]$  could be represented as  $[-1, 1] * (b-a)/2 + (a+b)/2$ . Any arbitrary closed-open interval  $[a, b)$  could be represented as  $[0, 1) * (b-a) + a$ . Any arbitrary infinite interval  $[a, \text{inf})$  could be represented as  $[0, \text{inf}) + a$ .

Also the UNION operator can be used to represent disjoint intervals. For example, the solution of the inequality

$$(x + 1) \cdot (x - 1) \cdot (x - 2) > 0$$

could be displayed as

$$x = \text{epsilon}((-1, 1) \text{ UNION } (2, \text{inf}))$$

Although cumbersome, introducing new data structures to represent arbitrary intervals directly AND algebraically simplifying them would be a huge project, considering the work required just to implement the simplification of expressions involving  $\#Z$  and  $\#R$ .

I hope you don't mind me using you as a sounding board. Please do not feel obligated to respond, but I will appreciate your comments if you do.

Aloha, Albert

**From:** [David Jeffrey](#)

**To:** [Albert D. Rich](#)

**Cc:** [David Stoutemyer](#) ; [Michel Beaudin](#) ; [Bernhard Kutzler](#)

**Sent:** Monday, 12 September, 2005 04:01

Aloha  $\text{epsilon}\{\text{Albert}, \text{David}\}$ ,

I think the notation is a great idea. The standard notation  $x : \text{epsilon } \#R$  becomes an abbreviation for  $x := \text{epsilon } \#R$ .

One notational question that still needs addressing is that of multiple instances of the expressions. For example,  $x := \text{epsilon}\{+1, -1\}a - \text{epsilon}\{+1, -1\}b$ .

Does this imply the 4 quantities  $a+b, a-b, -a+b, -a-b$ , or only the two quantities  $a-b, -a+b$ ?

With regard to intervals, my first comment is that implementing  $\text{epsilon}$  will keep you busy for enough cold Hawaiian nights huddled around the fireplace. However, for reference, I checked to see what Maple syntax is: they have a function  $\text{RealRange}(a, b)$  which is closed unless the end point is enclosed in another operator  $\text{Open}(a)$ . Thus the interval  $(a, b]$  is represented  $\text{RealRange}(\text{Open}(a), b)$

All for now: classes to teach

David

p46	From Multivalued Constants to Infinite Sets, a Report	D-N-L#59
-----	---	----------

**From:** [Stoutemyer, David](#)

**To:** [David Jeffrey](#) ; [Albert D. Rich](#)

**Cc:** [Michel Beaudin](#) ; [Bernhard Kutzler](#)

**Sent:** Monday, 12 September, 2005 07:19

Hi Albert,

I too like the unary epsilon idea and the generalization of union to arbitrary elements of sets.

Regarding the user's perspective of math syntax in general and more specifically interval syntax, it is way less missionary work to stick as close as possible to the most commonly used notation. One way to avoid syntactic ambiguity with vectors and function application is to separate the endpoints of an interval with an ellipsis rather than a comma. For example,  $[3\dots 5]$  or  $[3\dots 5)$ . This also emphasizes that it is a continuous range rather than merely the two endpoints, such as in the set notation  $\{3, 5\}$ .

Regarding internal representation of an interval, Derive already has (INTERVAL ...) and (STRICT ...) analogous to Maples RealRange(...) and Open(...). This seems more direct than using sin(...), mod(...) and abs(...), particularly since INTERVAL is already there and STRICT gives more direct control over the endpoint type. Also, why not atan(...) or tanh(...) in place of sin(...), etc.? It somehow feels wrong to have to go all the way to these complicated functions to represent something so fundamental as an interval.

-- keep up the good work!

David S.

**From:** [Albert D. Rich](#)

**To:** [David Jeffrey](#) ; [David Stoutemyer](#)

**Cc:** [Michel Beaudin](#) ; [Bernhard Kutzler](#)

**Sent:** Tuesday, 13 September, 2005 03:07

Hello David^2,

Thanks for your comments. In response to David Jeffrey's comments and suggestions:

1. No, the :epsilon operator is not equivalent to :=epsilon. The expression  $x : \text{epsilon } \#R$  *declares*  $x$  to be real, but  $x$  will still simplify to  $x$ . On the other hand, the expression  $x := \text{epsilon } \#R$  *assigns*  $x$  the value of an arbitrary element of the reals, so  $x$  will actually simplify to the multivalued expression  $\text{epsilon } \#R$ .
2. Multiple occurrences of multivalued constants in expressions are entirely uncorrelated (i.e. they do not necessarily have the same value). If you start with two independent expressions each having multivalued constants and then add them, why should the multivalued constants magically become correlated?

Naturally, great care must be taken when simplifying expressions having uncorrelated multivalued constants. For example, if  $C$  is a multivalued expression,  $C \cdot (a+b)$  is *not* equivalent to  $C \cdot a + C \cdot b$  (i.e.  $C$  cannot be distributed over or factored from a sum). More specifically,  $C-C$  is not necessarily 0! In general, before applying transformation rules that result in an expression being repeated, as well as the converse of such transformations, a check must be made to ensure that the expression is NOT multivalued.

This, along with the requirements of display stepping, has forced me to move away from the philosophy of a CAS having an absolute minimal set of primitive functions. Instead a richer set is required to avoid the repetition of expressions. For example,  $\text{TAN}(z)$  should *not* be replaced internally by  $\text{SIN}(z)/\text{COS}(z)$  (as it is in Derive 6), since it results in two occurrences of  $z$  and is thus invalid if  $z$  is a multivalued expression. Similarly,



$\text{ASIN}(z)$  should *not* be replaced by  $\text{ATAN}(z/\text{SQRT}(1-z^2))$ ,

$\text{ABS}(z)$  should *not* be replaced by  $z/\text{SIGN}(z)$ , and

$\text{MOD}(z)$  should *not* be replaced by  $z-\text{FLOOR}(z)$ .

Unfortunately, each additional primitive function results in an exponential growth in the number of transformation rules a CAS requires. Sigh...

In response to David Stoutemyer's comments and suggestions:

1. Using the ellipsis along with square brackets and parentheses to designate intervals is an excellent idea! Note that Derive currently allows the ellipsis to be used to enter vectors. For example,  $[1, \dots, 5]$  is a quick way to enter the vector  $[1, 2, 3, 4, 5]$ . Fortunately, however, the closed interval  $[1 \dots 5]$  is distinguishable from the vector by not having commas around the ellipsis.
2. The internal Derive utility function  $\text{INTERVAL}$  and the data structure it produces have little relevance to the problem of providing users with arbitrary intervals (i.e. infinite sets) that can be simplified as first class objects. For one thing, the user can enter symbolic and/or complex endpoints for intervals. For example, the user can enter the interval  $(- \#i \dots \#i)$  meaning all the points on the complex axis between  $\#i$  and  $- \#i$ .  $\text{INTERVAL}$  only handles numeric and infinite endpoints.
3. I agree that intervals are fundamental and storing them internally as  $\{\text{SIN}(\text{epsilon}(\#R))\}$ ,  $\{\text{MOD}(\text{epsilon}(\#R))\}$ , etc. is too cumbersome. You propose some sort of internal interval data structure with slots for storing the endpoints and strictness. This would result in possibly complicated symbolic endpoints being hidden in this data structure, requiring a whole new simplification system.

Instead, I propose five new constants for the intervals  $[0 \dots 1]$ ,  $(0 \dots 1)$ ,  $[0 \dots \text{inf})$ , and  $(0 \dots \text{inf})$ . Note that Derive already has the constant  $\#R$  for the interval  $(-\text{inf} \dots \text{inf})$ . Then when the user enters an interval, it can be stored internally as follows:

$[a \dots b]$  stored as  $[0 \dots 1] * (b-a) + a$

$(a \dots b)$  stored as  $(0 \dots 1) * (b-a) + a$

$[a \dots b)$  stored as  $[0 \dots 1) * (b-a) + a$

$(a \dots b]$  stored as  $[0 \dots 1) * (a-b) + b$

$[a \dots \text{inf})$  stored as  $[0 \dots \text{inf}) + a$

$(a \dots \text{inf})$  stored as  $(0 \dots \text{inf}) + a$

$(-\text{inf} \dots a]$  stored as  $-[0 \dots \text{inf}) + a$

$(-\text{inf} \dots a)$  stored as  $-(0 \dots \text{inf}) + a$

Then Derive's existing simplification machinery will automatically simplify expressions involving intervals. For example, the intervals  $2*[a \dots b)$  and  $(2*b \dots 2*a]$  will both simplify to the same internal form  $[0 \dots 1) * 2 * (b-a) + 2*a$ , and are thus equal intervals.

If the leading coefficient of either of the intervals  $[0 \dots 1]$  or  $(0 \dots 1)$  is negative, the normalization rules

$$-n*[0 \dots 1] \rightarrow n*[0 \dots 1] - n$$

$$-n*(0 \dots 1) \rightarrow n*(0 \dots 1) - n$$

are applied to make the leading coefficient positive. Therefore,  $[a \dots b]$  and  $[b \dots a]$  will both simplify to the same internal form, and are thus equal intervals.

As most recipients of this email already know, my consulting agreement with Texas Instruments Incorporated recently expired. Therefore whether these ideas actually make it into a future release of Derive is an open question.

Anyway, thanks again for your comments.

Aloha,

Albert

**From:** [Josef Böhm](#)

**To:** [Albert D. Rich](#)

**Sent:** Monday, 12 September, 2005 21:05

Hi Albert,  
exciting exchange of mails.  
Are there any problems in publishing them in the next DNL.

Best regards

Josef

**From:** [Albert D. Rich](#)

**To:** [Josef Böhm](#)

**Sent:** Monday, 13 September, 2005 10:011

Hello Josef,

Yes, this interaction of ideas has been an valuable learning experience for me. It brings to light how my thinking evolves in response to the dictates of mathematics and the comments of real world users like David's Stoutemyer and Jeffrey. I have no problems with you publishing them in the DNL.

Aloha,

Albert

**From:** [David Jeffrey](#)

**To:** [Albert D. Rich](#)

**Cc:** [David Stoutemyer](#) ; [Michel Beaudin](#) ; [Bernhard Kutzler](#) ; [Josef Böhm](#)

**Sent:** Tuesday, 13 September, 2005 12:13

Dear All,

1. Names.

Strictly the symbol in question is not an epsilon.

I have attached two expressions, printed in Knuth's Latex system, showing "x epsilon #R" and "x is-a-member-of #R".

Now that Derive has unicode, and can distinguish between "i" as a variable and "i" as sqrt(-1), it should also distinguish between epsilon and member-of. I am sure Knuth would disapprove of using epsilon. It would be the same as replacing PRODUCT and SUM with PI and SIGMA because the operators look like the letters.

<b>D-N-L#59</b>	<b>From Multivalued Constants to Infinite Sets, a Report</b>	<b>p49</b>
-----------------	--	------------

## 2. Correlation.

The expression  $1/(a \pm vb) - (a - \pm vb)/(a^2 - b)$

reduces to zero if the two " $\pm$ " operators are correlated, but not otherwise. The question is whether a system should offer this possibility to the users.

In Abramowitz and Stegun, we see on page 80 the formula

$$\text{Arctan } z_1 \pm \text{Arctan } z_2 = \text{Arctan}((z_1 \pm z_2)/(z_1 z_2 \mp 1))$$

Notice the " $\mp$ " symbol. Should this be possible ?

Michel, do you have some examples from your differential equations where the same arbitrary constant of integration appears in numerator and denominator?

David

**From:** [Michel Beaudin](#)

**To:** [David Jeffrey](#)

**Cc:** [David Stoutemyer](#) ; [Bernhard Kutzler](#) ; [Josef Böhm](#)

**Sent:** Tuesday, 13 September, 2005 23:19

Hello my friends.

For David Jeffrey's question to M.Beaudin: consider the separable --non linear-- ODE  $dy/dt = (y-2)^2$ . Then the general solution is  $y = 2 - 1/(t+c)$  where  $c$  is an arbitrary constant (and  $y = 2$  is another solution that does not follow from a particular choice of  $c$ ). If you put the RHS of the answer over the same denominator, you get  $y = (2t+2c-1)/(t+c)$ . Is it what you are looking for?

Another question to All: how can we verify the formula (page 80 of Abramowitz and Stegun) with Derive?. In Derive, I type the following expression

$$\text{atan}(x) + \text{atan}(y) - \text{atan}((x+y)/(1-x*y)).$$

Simplifying gives  $\text{atan}(x) + \text{atan}(y) + \text{atan}((x+y)/(x*y-1))$ , which is the same (atan is odd). And, plotting the graph (using the slide bar with  $y$  as a parameter), gives, depending on  $y$ , 2 horizontal segments. So, the equality seems to be valid only for a certain range.

Best regards.

Michel Beaudin

**From:** [David Jeffrey](#)

**To:** [Albert D. Rich](#)

**Cc:** [David Stoutemyer](#) ; [Michel Beaudin](#) ; [Bernhard Kutzler](#) ; [Josef Böhm](#)

**Sent:** Tuesday, 13 September, 2005 23:35

Michel,

The relation is not valid for all  $x,y$ . The correct relation between the arctangents requires a step function in the real case, or an unwinding number in the complex case. I introduced the example to show correlated "+" - signs, but otherwise the example is not relevant to the discussion.

Well in a sense it is. The lazy way to repair the relation is to add an unknown multiple of  $\pi$ , which could be done with Albert's new construction, but the multiple of  $\pi$  required is computable in this case from  $x,y$ , and that is a better solution for this case.

David

**From:** [Albert D. Rich](#)

**To:** [Josef Böhm](#)

**Sent:** Thursday, 15 September, 2005 08:27

Hello Josef,

Thank you for your interest in publishing this exchange of emails showing how the syntax and semantics of a significant addition to Derive capabilities is arrived at through the candid exchange of ideas between users and developers.

In my initial email I proposed some multivalued constants (i.e. #Z, #R, and #Un) that would greatly simplify the solutions returned for some equations and inequalities. Due to constructive comments from David<sup>2</sup>, this evolved into providing an arbitrary element operator for use on finite sets and the infinite sets script Z and R. Finally, it became apparent that primitives for the unit length intervals (i.e. infinite sets of reals) was also required for completeness.

Therefore, the title of this exchange of emails should probably not be "The benefit of multivalued constants". Rather something more broad, encompassing the arbitrary element operator, infinite sets, and intervals.

Aloha,

Albert

**From:** [Albert D. Rich](#)

**To:** [David Jeffrey](#) ; [David Stoutemyer](#)

**Cc:** [Michel Beaudin](#) ; [Bernhard Kutzler](#) ; [George Douros](#)

**Sent:** Thursday, 15 September, 2005 10:55

Hello All,

Thanks for this lively discussion on what I thought was an esoteric topic of interest to few.

1. I made a mistake proposing to use the lower case Greek letter epsilon to denote the arbitrary element (member) operator. epsilon should remain available for use as a variable name and distinct from the operator. Thanks to the foresight of George Douros, the Derive Unicode character set already has a distinct character for displaying the arbitrary element operator. I plan to use ELEMENT as the equivalent ASCII name for the operator.

2. Concerning the issue of correlation: DJJ gives the formula on page 80 of Abramowitz and Stegun as an example of the need for correlated "+-" operators. I feel the formula is just a way to show to two identities as one equation, thereby saving some space on the page. I have yet to see a problem for which a CAS would need correlation to show the solution.

Also an arbitrary element of the set {1, -1} (i.e. "+-1") is a simple multivalued expression for which one can at least contemplate correlation. However, no one would expect two occurrences of arbitrary elements of more complicated sets (e.g. the set of all reals or intervals) to be correlated. If correlation is required, use a variable with an appropriately declared domain. For example, the familiar identity

$$\sin(x)^2 + \cos(x)^2 = 1$$

would be confusing if written as

$$\sin(\text{ELEMENT}(\#R))^2 + \cos(\text{ELEMENT}(\#R))^2 = 1$$

with the assumption that the ELEMENT(#R)s are correlated.

3. The 5 primitive intervals proposed in my last email can and should be reduced to just the 3 unit length intervals by eliminating the 2 infinite length intervals  $[0...inf)$  and  $(0...inf)$ . Then intervals of the form  $[a...inf)$  can be replaced with  $[0...1)*inf + a$ , and intervals of the form  $(a...inf)$  can be replaced with  $(0...1)*inf + a$ .

Aloha,

Albert

**From:** [David Jeffrey](#)

**To:** [Albert D. Rich](#)

**Cc:** [Michel Beaudin](#) ; [Bernhard Kutzler](#) ; [David Stoutemyer](#) ; [George Douros](#)

**Sent:** Thursday, 15 September, 2005 14:49

With regard to 2. in Albert's mail:

In Michel's example of an ODE whose solution had the same constant in numerator and denominator, it appeared that the constant only appeared in 2 places because Michel put the answer over a common denominator.

Hence it looks as though a CAS should be able to get every answer in a form in which the constant only appears once. However, I am sure that if Michel looks around a little longer, he can find us an example in which it is not possible to put the answer in such a form. Even if he cannot find an example right away, there remains the possibility that a CAS would also put the expression over a common denominator. Perhaps the best way to handle correlation if it is needed is to introduce an auxiliary variable. To create a quick example:

SOLUTIONS([x + y = 1], [x, y])

simplifies at present to

[[@2, 1 - @2]]

where at present Derive does not state @2 : ELEMENT #R. If it did, however, then all occurrences of @2 would have the same. If however this were replaced with

[[ELEMENT(#R), 1- ELEMENT(#R)]]

then the two objects would be uncorrelated.

DJJ

**From:** [Albert D. Rich](#)

**To:** [David Jeffrey](#) ; [David Stoutemyer](#)

**Cc:** [Michel Beaudin](#) ; [Bernhard Kutzler](#) ; George Douros

**Sent:** Sunday, 25 September, 2005 07:57

**Subject:** Infinite sets and intervals

Hello All,

At the end of my email dated 12 September 2005 below, I propose representing general intervals internally using 3 primitive unit interval constants multiplied by a scale factor and added to by an offset. The alternative is to have an interval function having as arguments the bounds and strictness. The reason I give for using unit interval constants was that "Derive's existing simplification machinery will automatically simplify expressions involving intervals."

p52	A Call for Ideas	D-N-L#59
-----	------------------	----------

Now I have more rationale for my belief: The complex number  $x+y\cdot\#i$  can be represented by a imaginary unit constant  $\#i$  multiplied by  $y$  and added to by  $x$ . The alternative is to have a REIM function having as arguments  $x$  and  $y$ . Unfortunately, using REIM forces a CAS to properly handle two forms of sums and two forms of products all over the place! For this reason, David Stoutemyer and I have been driven to the conclusion that using REIM instead of an imaginary unit constant is significantly more difficult to implement.

The choice between using unit interval constants instead of an interval function is entirely analogous (isomorphic?) to using an imaginary unit constant instead of a REIM function. Thus unit interval constants should be used. qed

Aloha, Albert

P.S. I took the liberty of changing the subject of this email exchange since the idea of "multivalued constants" has evolved into a much more ambitious plan to introduce an arbitrary element operator and support for infinite sets and intervals.

### Call for Ideas for a DERIVE - Competition

We'd appreciate ideas and comments from DUG members to help us devise a Derive competition. The competition has two aims:

1. To raise awareness of the benefits of using Derive amongst newcomers.
2. To encourage existing Derive users to make better educational use of Derive.

Therefore the competition will have two questions.

The question for newcomers should be one which could be answered by someone using the 30-day trial version, so we might stimulate new users to try Derive. Perhaps it might be something like "Create a Derive worksheet showing a graphical and algebraic approach to solving simultaneous equations."

The prize for this might be a free Derive single user licence.

The question for existing users, which could win them a School Site Licence Plus (or equivalent value licence) could be something like one of the following.

Project: Meaningful Use of Slider Bars (2D & 3D)

Project: Calculate in the Algebra Window and present in the 3D- or 2D-Plot-Window

Project: Combine CAS & Programming

Project: DERIVE & Voyage 200 - A Game for Two

Or perhaps a mathematical challenge.

Do any DUG members have suggestions for questions which would best achieve these aims? For the widest impact perhaps the level should be school level.

Cheers

Philip Yorke

**Chartwell-Yorke Ltd.**

114 High Street, Belmont Village, Bolton, Lancashire, BL7 8AL

England, tel (+44) (0)1204 811001, fax (+44) (0)1204 811008

info@chartwellyorke.com, www.chartwellyorke.com