MASARYKOVA UNIVERZITA
INNOVATION LECTURES (INNOIEC)
www.muni.cz

Binding and Kinetics for Experimental Biologists

Lecture 2

Evolutionary Computing: Initial Estimate Problem

Petr Kuzmič, Ph.D.
BioKin, Ltd.

WATERTOWN, MASSACHUSETTS, U.S.A.

Tento projekt je spolufinancován Evropským sociálním fondem a státním rozpočtem České republiky.

EVROPSKÁ UNIE · MINISTERSTVO ŠKOLSTVÍ, MLÁDEŽE A TĚLOVÝCHOVY · OP Vzdělávání pro konkurenceschopnost

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

BioKin Ltd.

Lecture outline

- The problem:

  Fitting nonlinear data usually requires an initial estimate of model parameters. This initial estimate must be close enough to the "true" values.

- The solution:

  Use a data-fitting method that does not depend on initial estimates.

- An implementation:

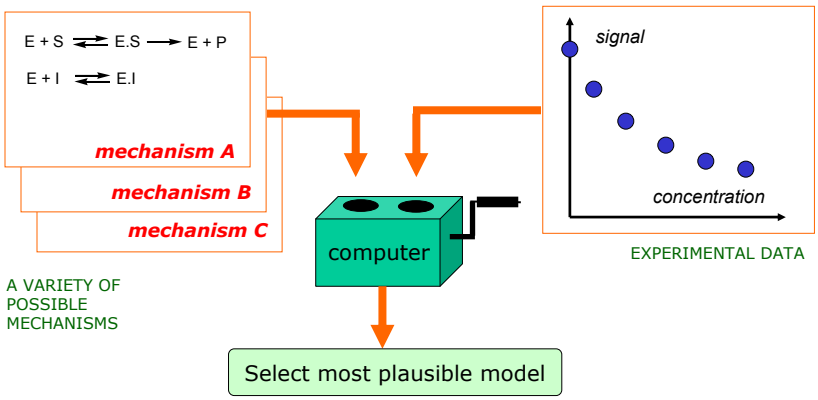
  The Differential Evolution algorithm (Price *et al.*, 2005).

- An example:

  Kinetics of forked DNA binding to the protein-protein complex formed by DNA-polymerase sliding clamp (gp45) and clamp loader (gp44/62).

The ultimate goal of analyzing kinetic / binding data

SELECT AMONG POSSIBLE **MOLECULAR MECHANISMS**

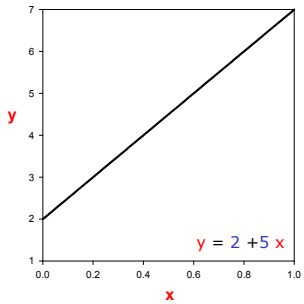


Most models in natural sciences are <u>nonlinear</u>

LINEAR VS. NONLINEAR MODELS

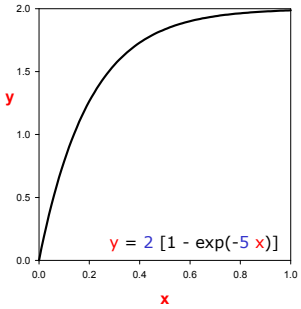
Linear

$y = A + k\,x$



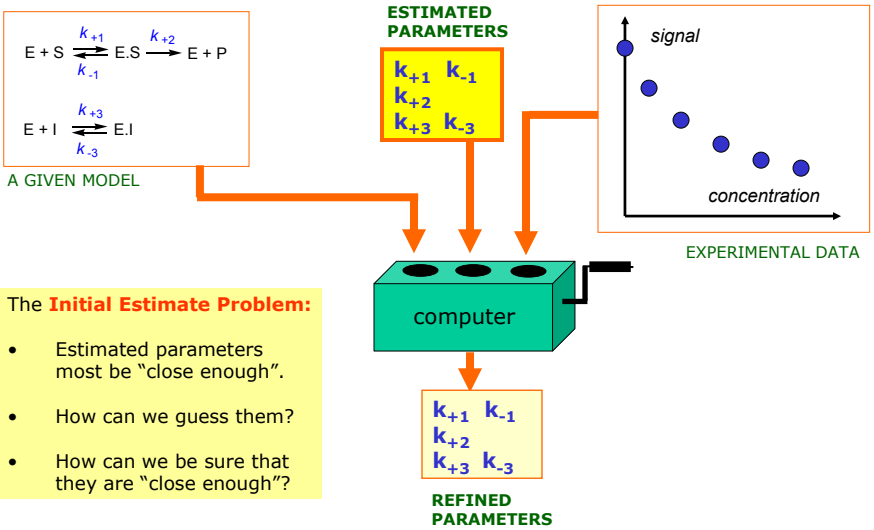
Nonlinear

$y = A\,[1 - \exp(-k\,x)]$



We need initial estimates of model parameters

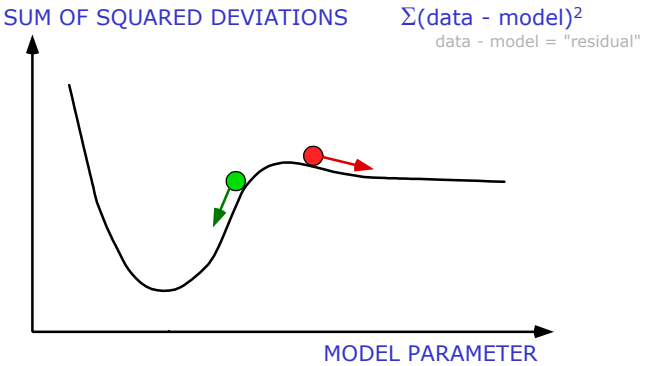
NONLINEAR MODELS REQUIRE INITIAL ESTIMATES OF PARAMETERS



The **Initial Estimate Problem:**

- Estimated parameters most be "close enough".
- How can we guess them?
- How can we be sure that they are "close enough"?

The crux of the problem: Finding *global* minima

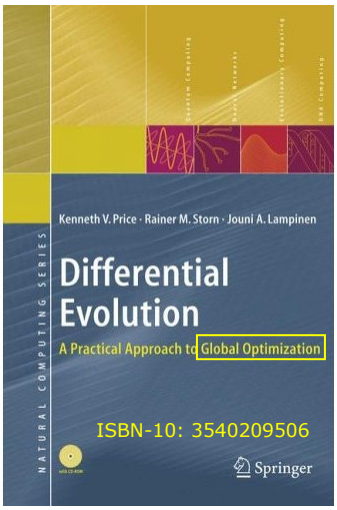


- Least-squares fitting **only** goes "downhill"
- **How do we know where to start?**

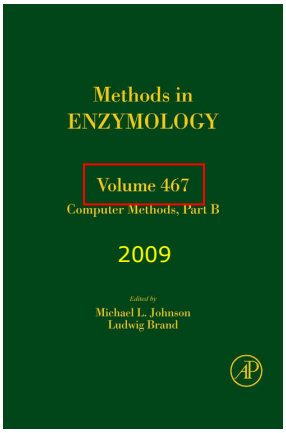
Charles Darwin to the rescue

BIOLOGICAL EVOLUTION IMITATED IN "DE"

Charles Darwin (1809-1882)



Specialized numerical software: *DynaFit*



CHAPTER TEN

DYNAFIT—A SOFTWARE PACKAGE FOR ENZYMOLOGY

Petr Kuzmič

DOWNLOAD http://www.biokin.com/dynafit

DynaFit implements the **Differential Evolution** algorithm for global sum-of-squares minimization.

Kuzmic (2009) *Meth. Enzymol.*, **467**, 247-280

Biological metaphor: "Gene, allele"

BIOLOGY	COMPUTER
gene	• sequence of bits representing a number
...AAGTCG...GTAACCGG... "keratin"	...0111001100001101101110011... "K_M" "k_{cat}"
four-letter alphabet variable length	• two letter alphabet • fixed length (16 or 32 bits)

"Chromosome, genotype, phenotype"

BIOLOGY	COMPUTER
genotype	• particular combination of all model parameters
...AAGTCGGTTCGGAAGTCGGTTTA...	011010110110 011110011010 001111101101
keratin oncoprotein	V_{max}=1.23 K_M=4.56 K_{is}=78.9
phenotype	full set of parameters

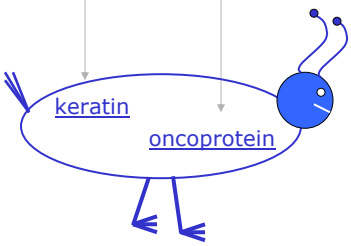
$$v = V_{\max} \frac{[S]/K_M}{1+[S]/K_M+[S]^2/K_M K_{is}}$$

"Organism, fitness"

BIOLOGY | COMPUTER

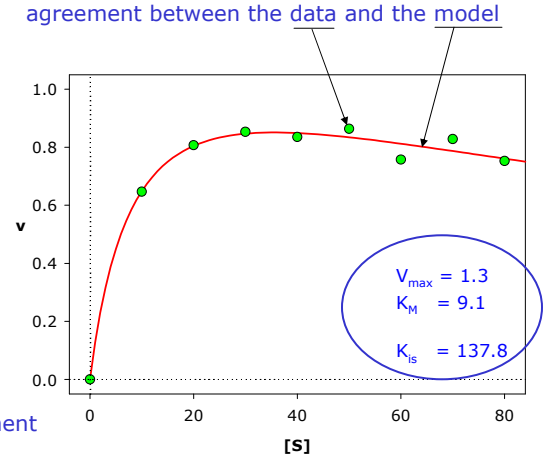
genotype

...AAGTCGGTTCGGAAGTCGGTTTA...



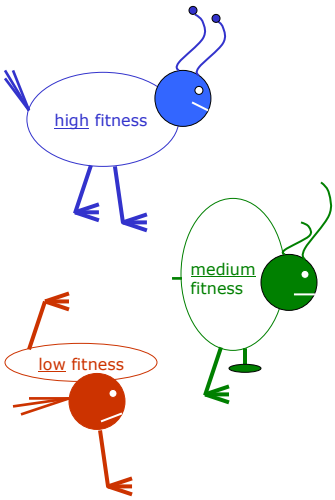
FITNESS:
"agreement" with the environment

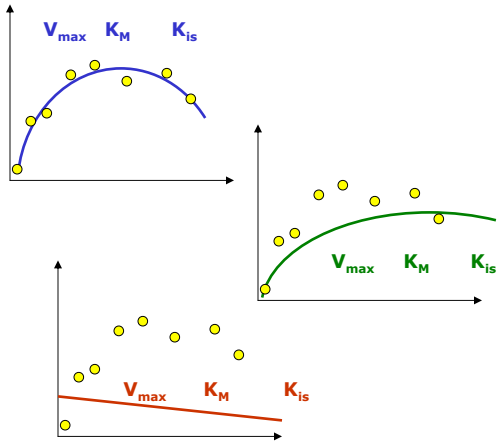
- FITNESS:
  agreement between the data and the model



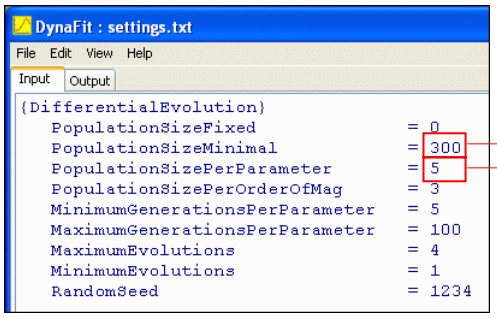
"Population"

BIOLOGY | COMPUTER





DE Population size in DynaFit

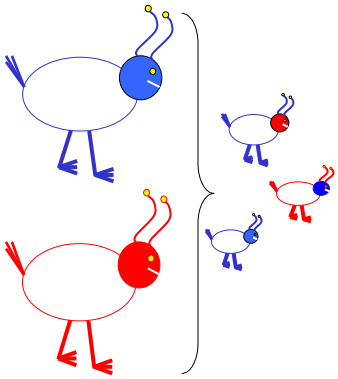


number of population members **per optimized model parameter**

number of population members **per order of magnitude**

"Sexual reproduction, crossover"

BIOLOGY | COMPUTER



random crossover point

mother
011010110110011110011 01 00011111011

father
011010110110011110011 01 11100011011

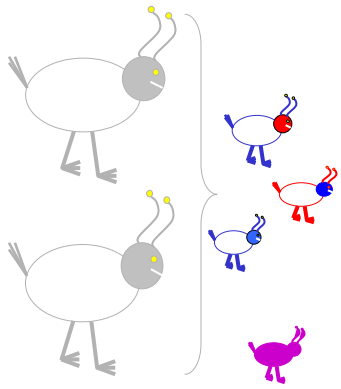
"sexual mating"
probability p_{cross}

child
011010110110 01111001101 11100011011

V_{max} K_M K_{is}

"Mutation, genetic diversity"

BIOLOGY | COMPUTER



father

01101011011	001111001101	11100011011
V_{max}	K_M	K_{is}

mutation

mutant father

11100111011	001011010101	11001011001
$V_{max}^{(*)}$	$K_M^{(*)}$	$K_{is}^{(*)}$

"Mutation, genetic diversity"

THE "**DIFFERENTIAL**" IN DIFFERENTIAL EVOLUTION ALGORITHM - **STEP 1**

Compute difference between two randomly chosen "auntie" phenotypes

aunt #1

01101011011	001111001101	11100011011
$V_{max}^{(1)}$	$K_M^{(1)}$	$K_{is}^{(1)}$

subtract

aunt #2

11100111011	001011010101	11001011001
$V_{max}^{(2)}$	$K_M^{(2)}$	$K_{is}^{(2)}$

aunt #2 **minus** aunt #1

11100111011	001011010101	11001011001
$V_{max}^{(2)}-V_{max}^{(1)}$	$K_M^{(2)}-K_M^{(1)}$	$K_{is}^{(2)}-K_{is}^{(1)}$

"Mutation, genetic diversity"

THE "**DIFFERENTIAL**" IN DIFFERENTIAL EVOLUTION ALGORITHM - **STEP 2**

Add **weighted** difference between two "uncle" phenotypes to "father"

father

01101011011	001111001101	11100011011
V_{max}	K_M	K_{is}

add a fraction of

aut #2 **minus** aut #1

11100111011	001011010101	11001011001
$V_{max}^{(2)}-V_{max}^{(2)}$	$K_M^{(2)}-K_M^{(1)}$	$K_{is}^{(2)}-K_{is}^{(1)}$

mutant father

11100111011	001011010101	11001011001
V_{max}^{*}	K_M^{*}	K_{is}^{*}

"Mutation, genetic diversity"

THE "**DIFFERENTIAL**" IN DIFFERENTIAL EVOLUTION ALGORITHM

EXAMPLE: Michaelis-Menten equation $v = V_{max}\frac{[S]}{[S]+K_M}$

"father" "aunt 1" "aunt 2"

"mutant father" → $K_M^{*} = K_M + \mathbf{F} \times (K_M^{(1)} - K_M^{(2)})$

weight (fraction)
mutation rate

DE "undocumented" settings in DynaFit

DynaFit : settings-HIDDEN.txt

File Edit View Help

Input Output

```
{DifferentialEvolution}
   CombineGenerations              = n
   ReplaceStragglersPercent        = 0
   Constrained                     = y
   Strategy                        = 3
   Weight                          = 0.8
   Crossover                       = 1
   Jitter                          = 0.01
   Distribution                    = uniform
   AddUserEstimate                 = n
   Scaling                         = logarithmic
   NormalDeviation                 = 0.5
   ExponentialLambda               = 2
   ReportFrequency                 = 1
   TestParameterRange              = y
   TestParameterRangeAll           = y
   TestParameterRangeFull          = n
   StopParameterRange              = 0.01
   TestCostFunctionRange           = y
   StopCostFunctionRange           = 0.000001
   TestCostFunctionChange          = y
   StopCostFunctionChange          = 0.000001
   TestCostFunctionChangeCount     = 10
```

six different mutation strategies

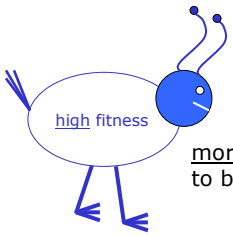
fractional difference used in mutations
$K_M^* = K_M + \mathbf{F} \times (K_M^{(1)} - K_M^{(2)})$

probability that **"child"** inherits **"father's"** genes, not **"mother's"** genes

These DE tuning constants are "undocumented" in the DynaFit distribution.

"Selection"

BIOLOGY	COMPUTER



more likely to breed

low sum of squares

01101011011	00111100110	00011111011
V_{max}	K_M	K_{is}

more likely to be carried to the next generation

less likely to breed

high sum of squares

00000000001	11111111111	00000000000
V_{max}	K_M	K_{is}

less likely to be carried to the next generation

Basic Differential Evolution Algorithm - Summary

1 **Randomly** create the initial population (size **N**)

Repeat until almost all population members have very high fitness:

2 Evaluate fitness: sum of squares for all population members

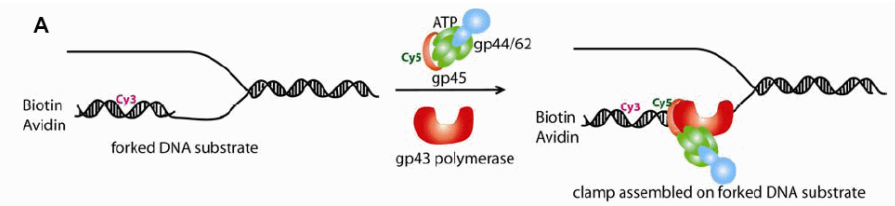
3 Mutation: **random** gene modification (mutate *father*, weight **F**)

4 Sexual reproduction: **random** crossover with probability P_{cross}

5 Natural selection: keep *child* in gene pool if more fit than *mother*

Example: DNA + clamp / clamp loader complex

DETERMINE ASSOCIATION AND DISSOCIATION RATE CONSTANT IN AN **A + B ⇔ AB** SYSTEM



Schematic representation of the clamp loading onto forked-DNA substrate. In the forked DNA-substrate, the primer carries a Cy3 fluorescent donor and the gp45 clamp contains an acceptor Cy5 dye.

see *Lecture 1* for details

Courtesy of Senthil Perumal, Penn State University (Steven Benkovic lab)

Example: DynaFit script for Differential Evolution

INSERT A SINGLE LINE IN THE [TASK] SECTION

DynaFit : fit-004.txt

File Edit View Help

Input Output

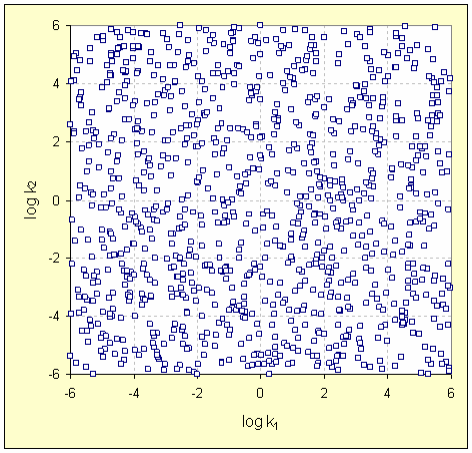
```
[task]

   task = fit
   data = progress
   algorithm = differential-evolution

[mechanism]

   DNA + Clamp.Loader <==> Complex   :     kon    koff

[constants]

   kon  = 1 ?
   koff = 1 ?
```

```
[responses]

   Complex = 1 ? (0.01 .. 100)

[data]

   file      ./courses/bkeb/lec-1/a+b/data/d1-edit.txt
   offset    0.3 ? (0.2 .. 0.4)
```

constraints !

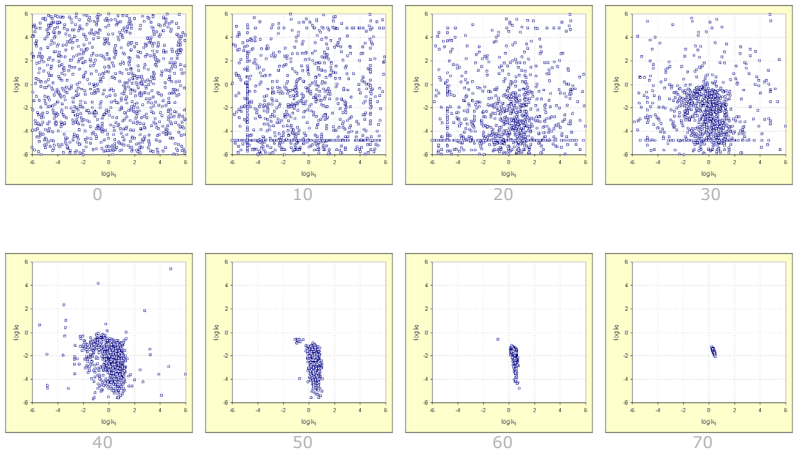
Example: Initial population

BOTH RATE CONSTANTS SPAN **TWELVE ORDERS OF MAGNITUDE**



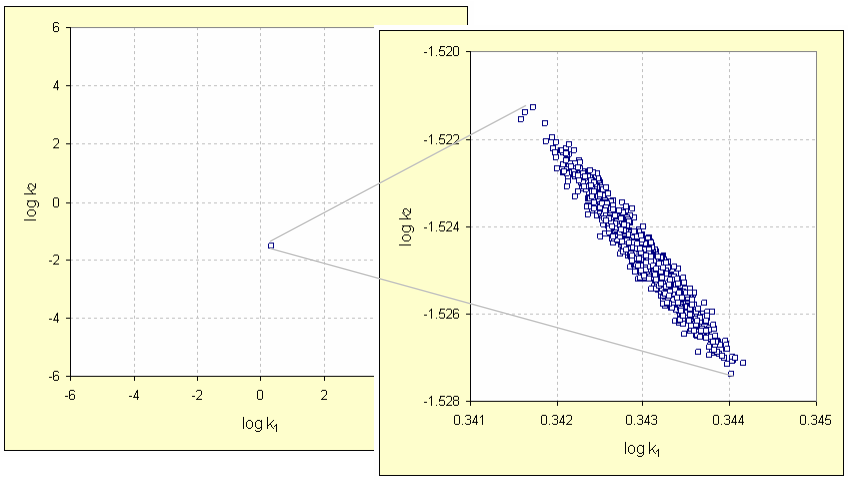
Example: The evolutionary process

SNAPSHOTS OF **k_1 / k_2 CORRELATION** DIAGRAM - SPACED BY 10 **"GENERATIONS"**



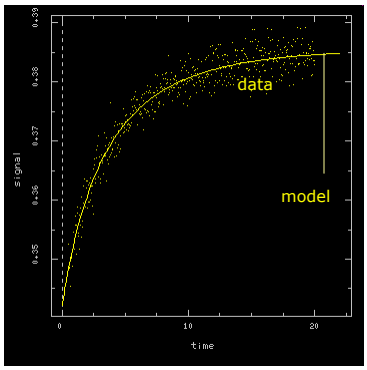
Example: Final population

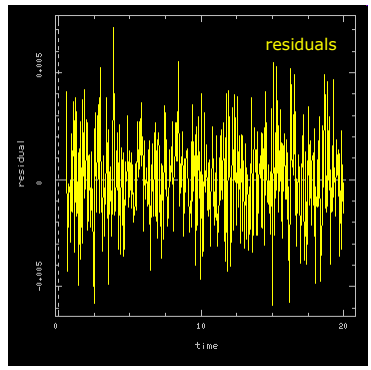
BOTH RATE CONSTANTS SPAN **AT MOST ±30% RANGE** RELATIVE TO NOMINAL VALUE



Example: The "fittest" member of final population

THIS IS (PRESUMABLY) THE **GLOBAL MINIMUM** OF SUM-OF-SQUARES





Regression Summary

Differential Evolution (EvolDEPSL)

sum of squares	0.00230769

Regression Summary

Levenberg-Marquardt Algorithm

sum of squares	0.00230769

compare with "good" estimate from Lecture 1

Example: Comparison of DE and regular data fitting

DIFFERENTIAL EVOLUTION (**DE**) FOUND THE **SAME FIT AS THE "GOOD" ESTIMATE**

		initial estimate	sum of squares	relative sum of sq.	"best-fit" constants
lecture 1	"good"	$k_1 = 1$ $k_2 = 1$	0.002308	1.00	$k_1 = 2.2 \pm 0.5$ $k_2 = 0.030 \pm 0.015$
lecture 1	"bad"	$k_1 = 100$ $k_2 = 0.01$	0.002354	1.02	$k_1 = 0.2 \pm 3.4$ $k_2 = 0.2 \pm 0.6$
1000 random estimates		$k_1 = 10^{-6} - 10^{+6}$ $k_2 = 10^{-6} - 10^{+6}$	0.002308	1.00	$k_1 = 2.2 \pm 0.5$ $k_2 = 0.030 \pm 0.015$

Significant disadvantage of DE: very slow

DYNAFIT CAN TAKE **MULTIPLE DAYS** TO RUN A **COMPLEX PROBLEM**

DynaFit 4.065 on DNA / clamp / clamp loader example:

algorithm	computation time	relative time
Levenberg-Marquardt with two restarts	0.88 sec	1
Differential Evolution with four restarts (population size: 1000)	12 min 31 sec	**853**

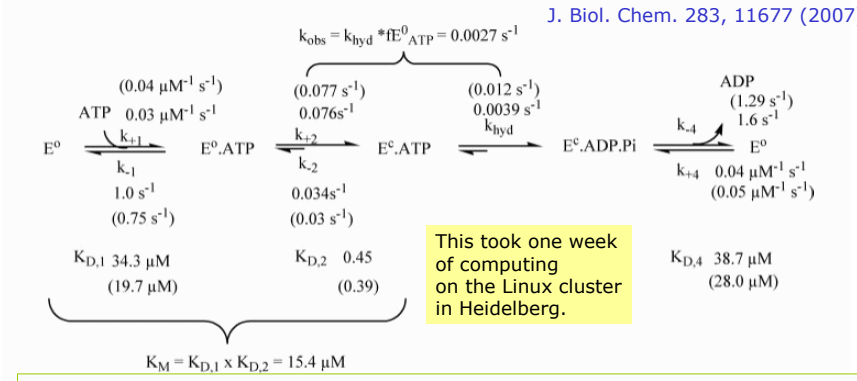
1 second
1 minute
10 minutes

→

15 minutes
15 hours
6 days

Example of Differential Evolution in DynaFit

J. Biol. Chem. 283, 11677 (2007)



This took one week of computing on the Linux cluster in Heidelberg.

THE JOURNAL OF BIOLOGICAL CHEMISTRY VOL. 283, NO. 17, pp. 11677–11688, April 25, 2008

The ATPase Cycle of the Mitochondrial Hsp90 Analog Trap1*

Received for publication, November 20, 2007, and in revised form, February 5, 2008 Published, JBC Papers in Press, February 20, 2008, DOI 10.1074/jbc.M709516200

Adriane Leskovar‡, Harald Wegele§1, Nicolas D. Werbeck‡, Johannes Buchner§, and Jochen Reinstein‡2
From the ‡Department of Biomolecular Mechanisms, Max-Planck-Institute for Medical Research, Jahnstrasse 29, Heidelberg 69120 and §Department of Chemie, Technische Universität München, Lichtenbergstrasse 4, Garching 85747, Germany

Example: Systematic scan of many initial estimates

CAREFUL! THIS IS FASTER THAN DIFFERENTIAL EVOLUTION BUT DOES NOT ALWAYS WORK

DynaFit : fit-007.txt

File Edit View Help

Input Output

```
[task]

   task = estimate
   data = progress

[mechanism]

   DNA + Clamp.Loader <==> Complex   :     kon    koff

[constants]

   kon  = {0.001, 0.01, 0.1, 1, 10, 100, 1000} ?
   koff = {0.001, 0.01, 0.1, 1, 10, 100, 1000} ?

[concentrations]

   DNA          = 0.1
   Clamp.Loader = 0.1
```

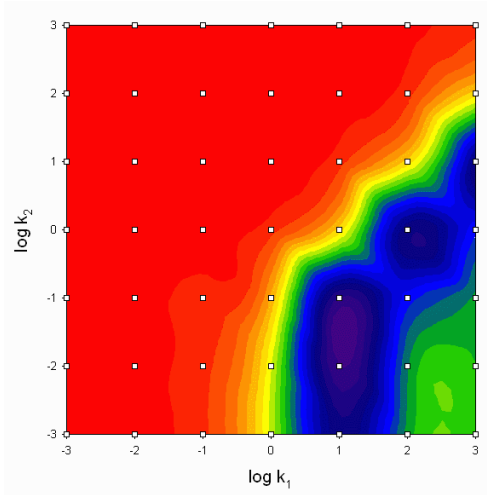
ALGORITHM

1. generate all possible combinations of rate constants
2. compute initial sum of squares for each combination
3. rank combinations by initial sum of squares
4. select the best **N** combinations
5. perform a full fit for those **N**
6. rank results again

$7 \times 7 = 49$ combinations of k_{on} and k_{off}

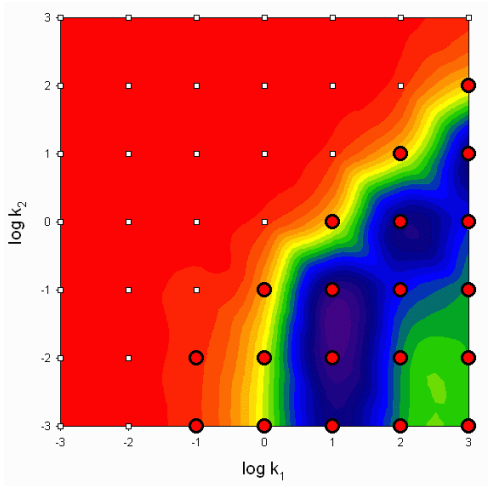
Example: Systematic scan – Phase 1

AFTER EVALUATING THE **INITIAL** SUM OF SQUARES FOR ALL 49 COMBINATIONS OF k_1 and k_2



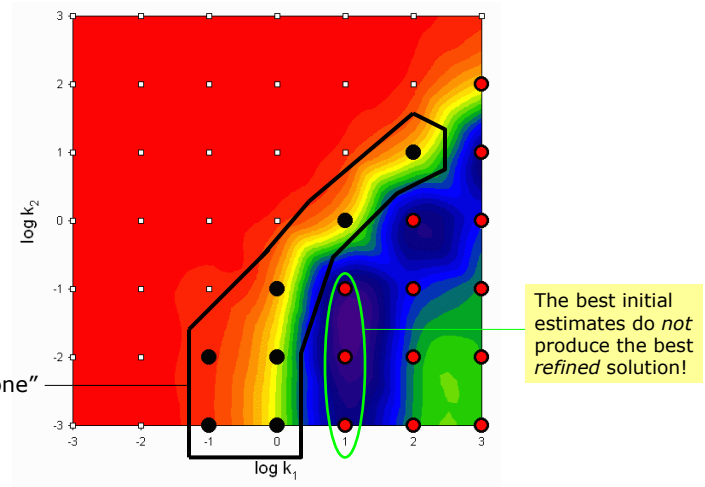
Example: Systematic scan – Phase 2

AFTER RANKING THE INITIAL ESTIMATES AND SELECTING **20 BEST ONES** BY SUM OF SQUARES



Example: Systematic scan – Phase 3

AFTER PERFORMING FULL REFINEMENT FOR **20 BEST ESTIMATES** OUT OF 49 TRIED



Summary and conclusions

1. Finding good-enough initial estimates is a very difficult problem.
2. One should use system-specific information as much as possible.
   This includes using the literature and/or general principles for "intelligent" guesses.
3. Always use the "Try" method in DynaFit to display the initial fit.
   Make sure that the initial estimate is at least approximately correct.
4. The Differential Evolution algorithm almost always helps.
   However, it can be excruciatingly slow (running typically for multiple hours).
5. The systematic scan (**`task = estimate`**) sometimes helps.
   However, the "best" initial estimates almost never produce the desired solution!
6. DynaFit is not a "silver bullet": You must still use your brain a lot.