Stochastic optimization algorithms
Lecture 4, 20200908

Evolutionary algorithms:
Background and introduction

Mattias Wahde, PhD, Professor, Chalmers University of Technology
e-mail: mattias.wahde@chalmers.se, http://www.me.chalmers.se/~mwahde

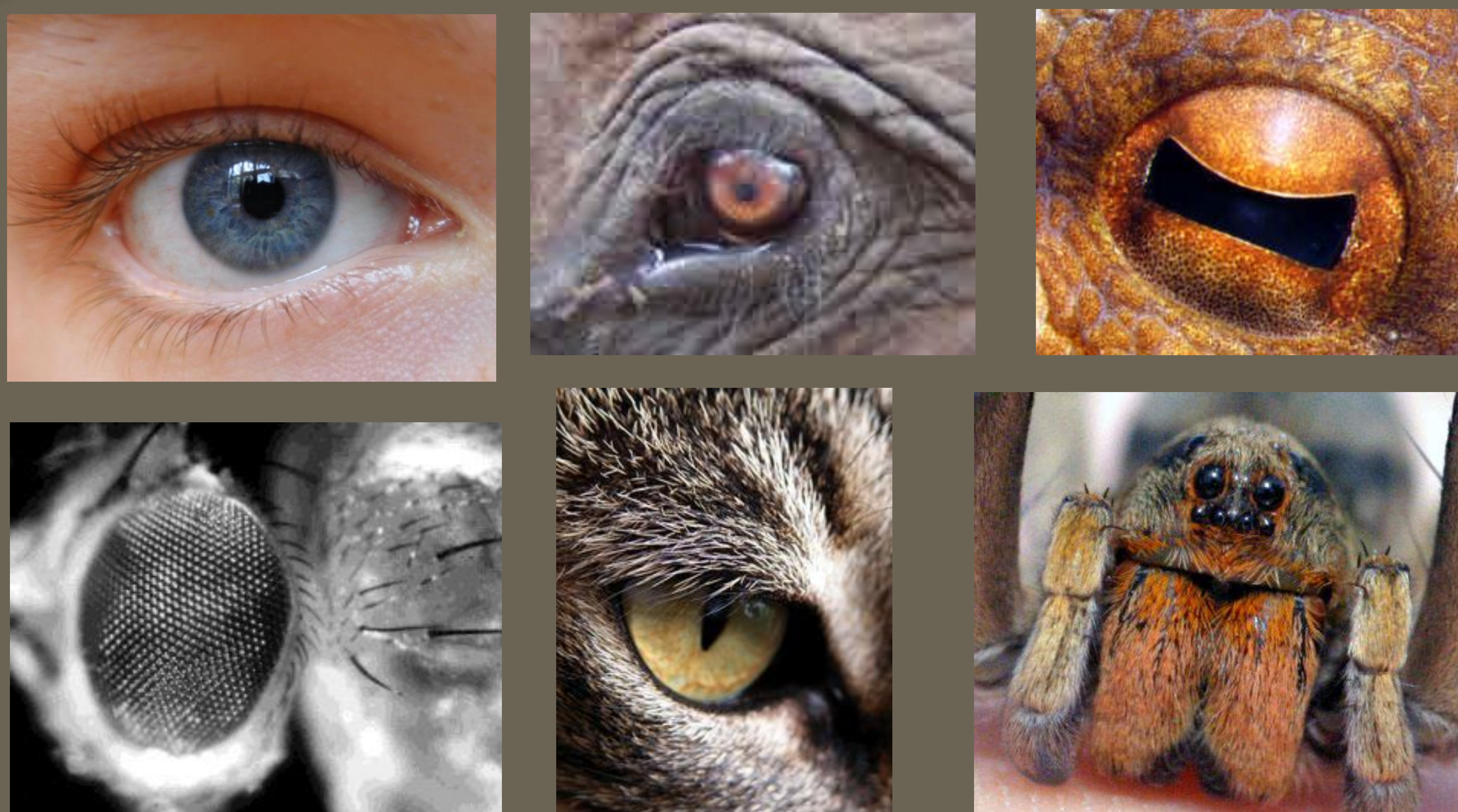
Today's learning goals

- After this lecture you should be able to
  - Motivate the use of evolutionary algorithms.
  - Describe the central concepts in Darwinian evolution.
  - Describe the structure and functionality of genes (the processes of transcription and translation).
  - Summarize the main aspects of evolution.
  - Describe the basic functionality of evolutionary algorithms.
  - Exemplify some of the simplifications used in evolutionary algorithms (relative to biological evolution).

Why evolutionary algorithms?

- Evolutionary algorithms ...
  - ...do not (usually) get stuck in local optima.
  - ...can find several different and equally viable solutions to a given problem.
  - ... are easily applicable to a wide variety of problems.
  - ... but also require some experience for successful usage.

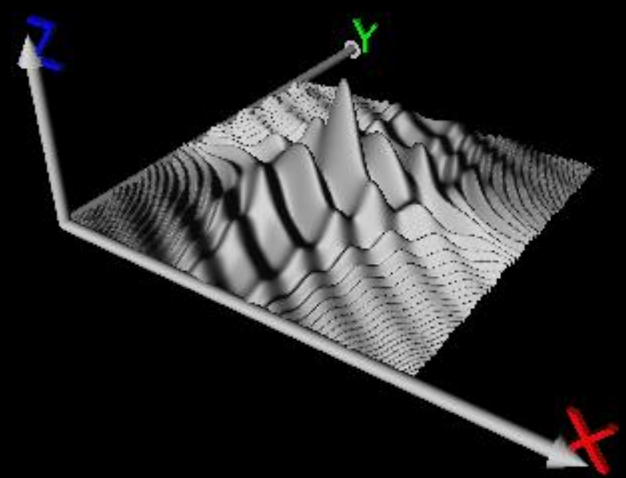
Example: Evolution of eyes



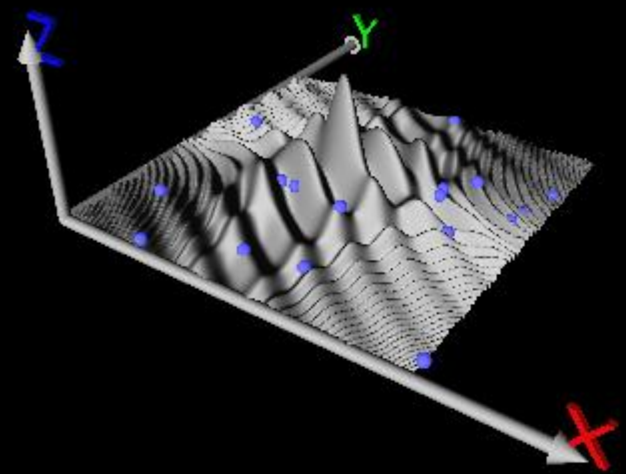
Mattias Wahde, PhD, Professor, Chalmers University of Technology
e-mail: mattias.wahde@chalmers.se, http://www.me.chalmers.se/~mwahde

CHALMERS

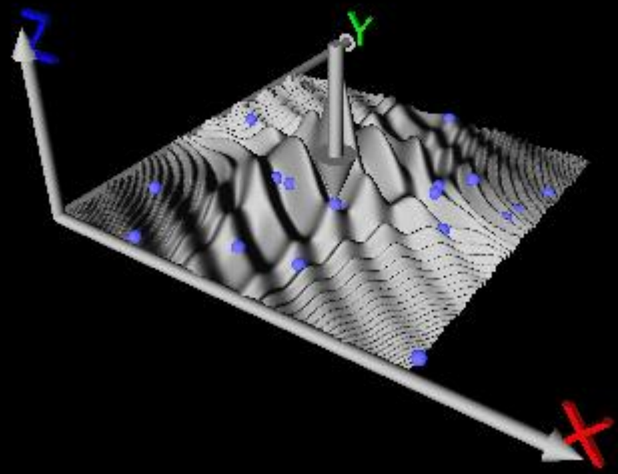
Demonstrations



Demonstrations

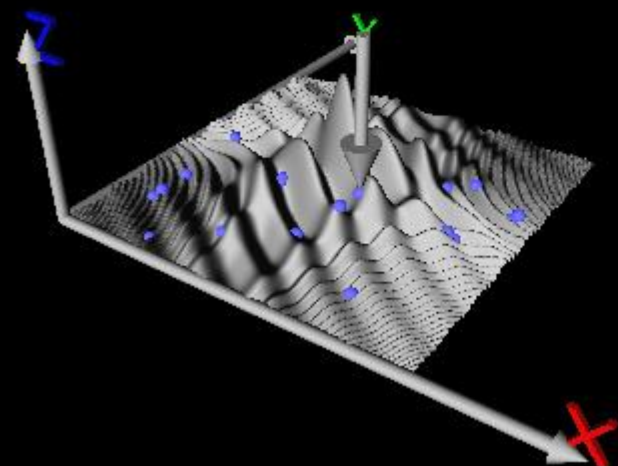


Demonstrations

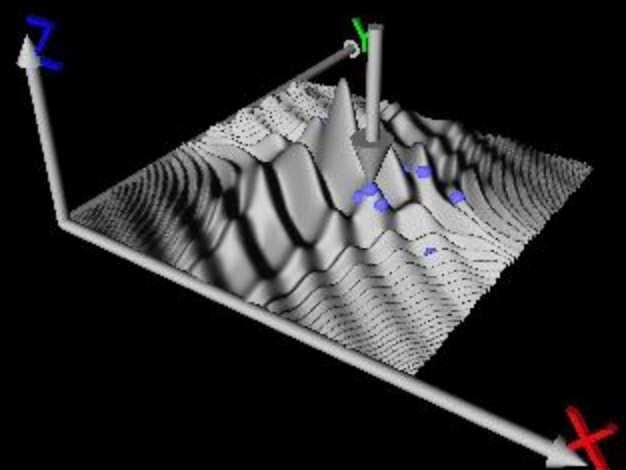


CHALMERS

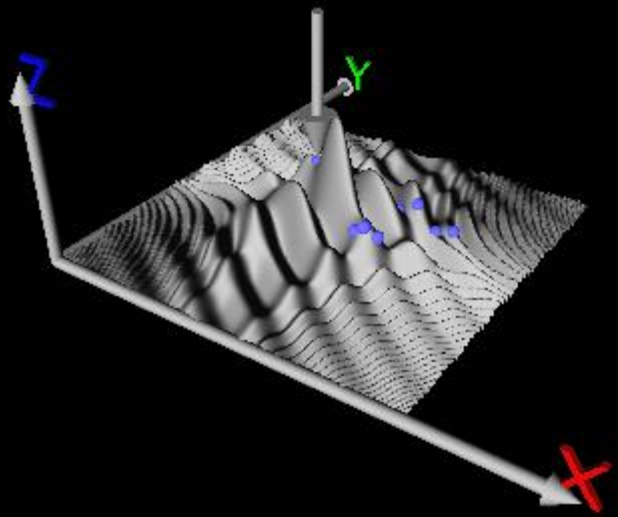
Demonstrations



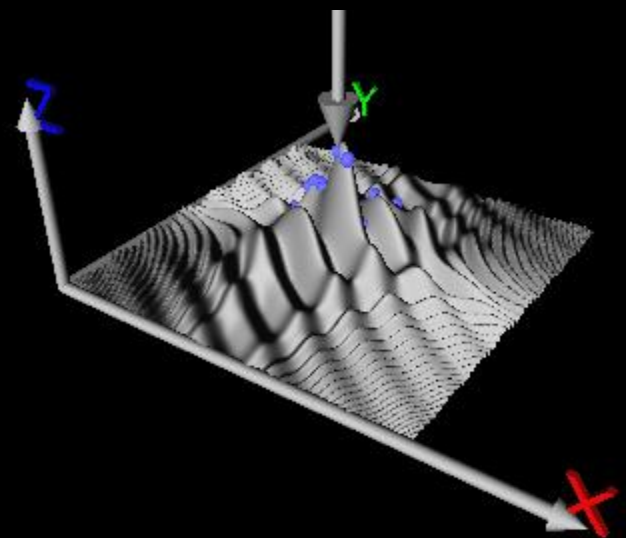
Demonstrations



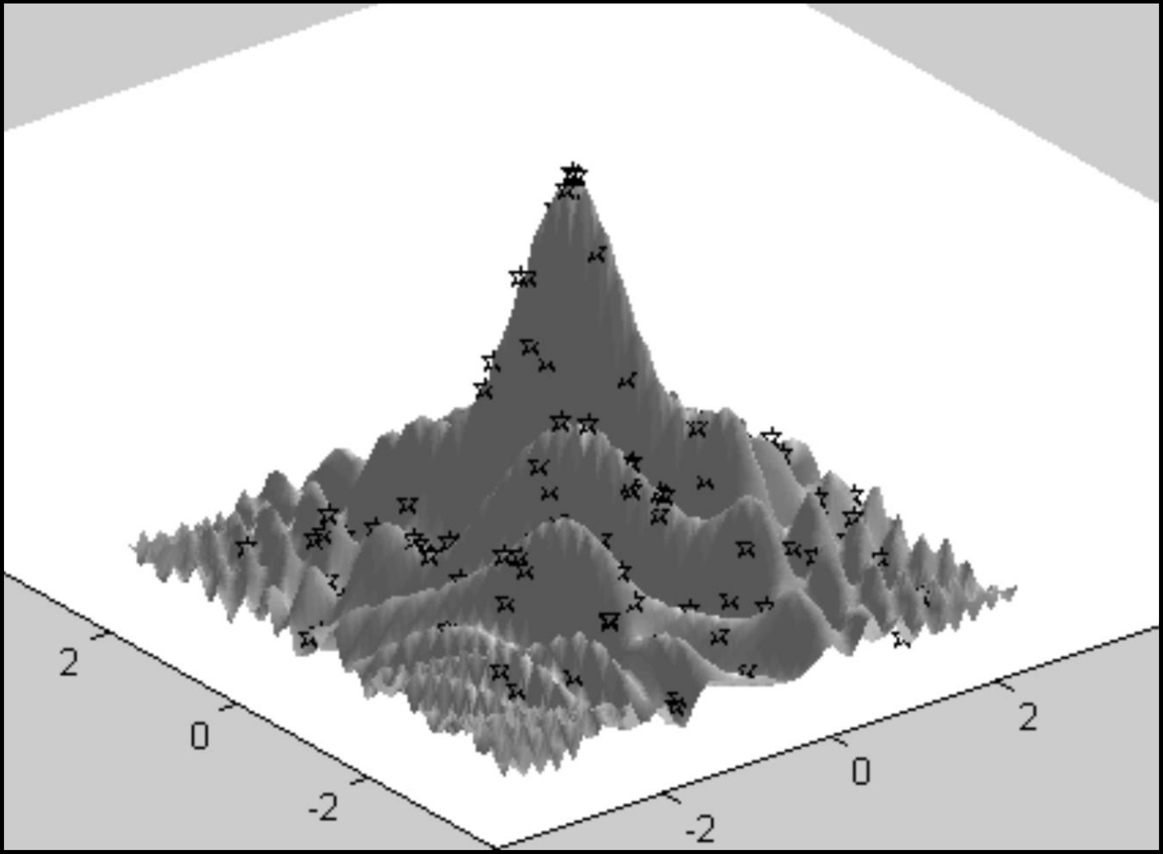
Demonstrations



Demonstrations



Demonstrations



Mattias Wahde, PhD, Professor, Chalmers University of Technology
e-mail: mattias.wahde@chalmers.se, http://www.me.chalmers.se/~mwahde

Today's learning goals

- After this lecture you should be able to
  - Motivate the use of evolutionary algorithms.
  - Describe the central concepts in Darwinian evolution.
  - Describe the structure and functionality of genes (the processes of transcription and translation).
  - Summarize the main aspects of evolution.
  - Describe the basic functionality of evolutionary algorithms.
  - Exemplify some of the simplifications used in evolutionary algorithms (relative to biological evolution).

Darwinian evolution

- Central concepts: Populations and individuals.
- A **population** is a group of **individuals** of the same **species**.
- Inidividuals of the same species can have fertile offspring.

pp. 35-39

Mattias Wahde, PhD, Professor, Chalmers University of Technology
e-mail: mattias.wahde@chalmers.se, http://www.me.chalmers.se/~mwahde

Liger = tiger + lion!

Darwinian evolution

- Central concept: Gradual, hereditary change
  - **Gradualness**: Changes occur over very long times.
  - **Heredity**: Properties of an individual can be stored so that they can be transmitted to the next generation (reproduction).

pp. 35-39

Mattias Wahde, PhD, Professor, Chalmers University of Technology
e-mail: mattias.wahde@chalmers.se, http://www.me.chalmers.se/~mwahde

Today's learning goals

- After this lecture you should be able to
  - Motivate the use of evolutionary algorithms.
  - Describe the central concepts in Darwinian evolution.
  - Describe the structure and functionality of genes (the processes of transcription and translation).
  - Summarize the main aspects of evolution.
  - Describe the basic functionality of evolutionary algorithms.
  - Exemplify some of the simplifications used in evolutionary algorithms (relative to biological evolution).

Darwinian evolution

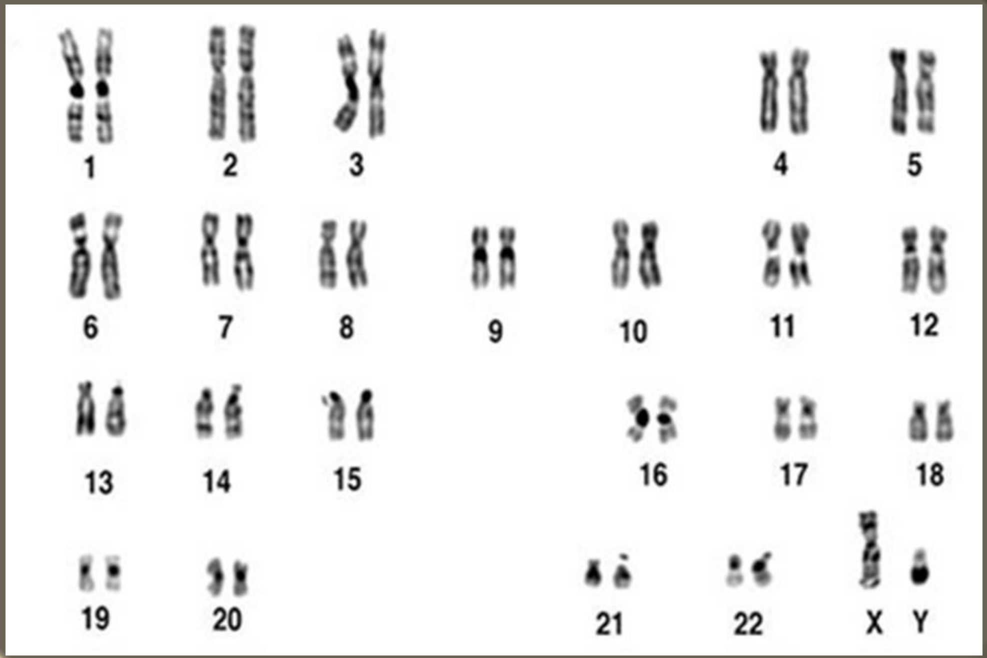
- How is the transferable information stored?
  - …in the **genome**, which (in humans) consist of 23 pairs of **chromosomes**.
  - Each chromosome, in turn, contains a large number of **genes**.

pp. 35-39

Mattias Wahde, PhD, Professor, Chalmers University of Technology
e-mail: mattias.wahde@chalmers.se, http://www.me.chalmers.se/~mwahde

Chromosome numbers

- Fruit fly : 8
- Cat: 38
- Human: 46
- Ape: 48
- Horse: 64
- Dog: 78
- Carp: 104



pp. 35-39

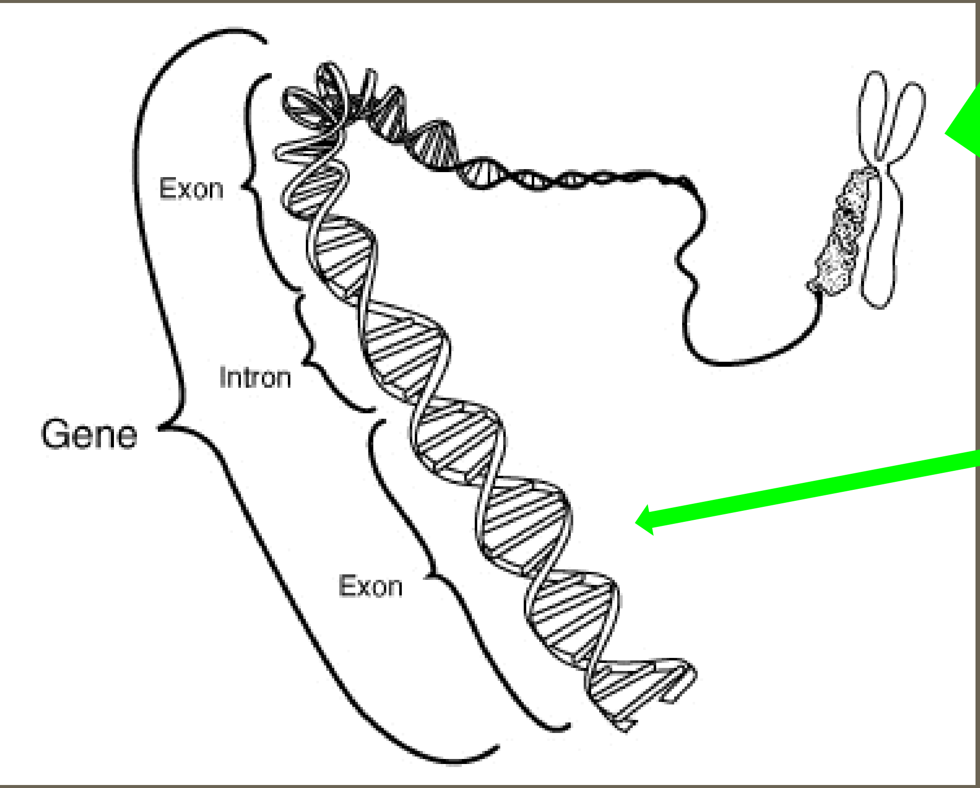
Number of genes

- Approximate number of genes in selected species:
  - Bacterium: 500 – 6000
  - Yeast: 6000
  - Fruit fly: 13600
  - Human: 25000
- However, there are many species (e.g. some fish) with more genes than humans.

pp. 35-39

Mattias Wahde, PhD, Professor, Chalmers University of Technology
e-mail: mattias.wahde@chalmers.se, http://www.me.chalmers.se/~mwahde

Chromosomes



Chromosome

Sequences of base pairs:

...A-C-C-T-C-G-G-T-C-A-G...

...T-G-G-A-G-C-C-A-G-T-C...

pp. 35-39

Mattias Wahde, PhD, Professor, Chalmers University of Technology
e-mail: mattias.wahde@chalmers.se, http://www.me.chalmers.se/~mwahde

Genes

- Genes consist of sequences of **base pairs**, i.e. pairs of molecules (whose names are abbreviated A, C, T, G).
- A is always paired with T, and C is always paired with G:

… A C T G G T C A A T G …

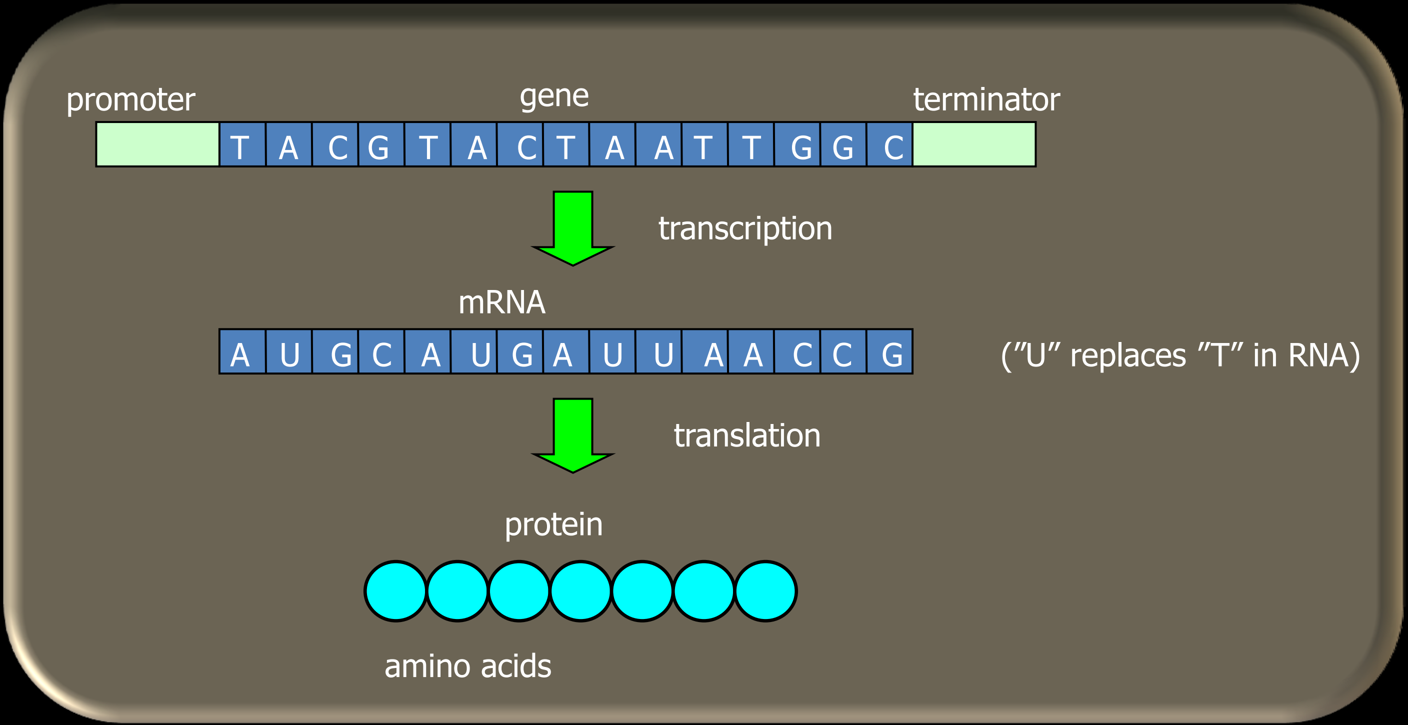
… T G A C C A G T T A C …

pp. 35-39

Mattias Wahde, PhD, Professor, Chalmers University of Technology
e-mail: mattias.wahde@chalmers.se, http://www.me.chalmers.se/~mwahde

The functionality of genes

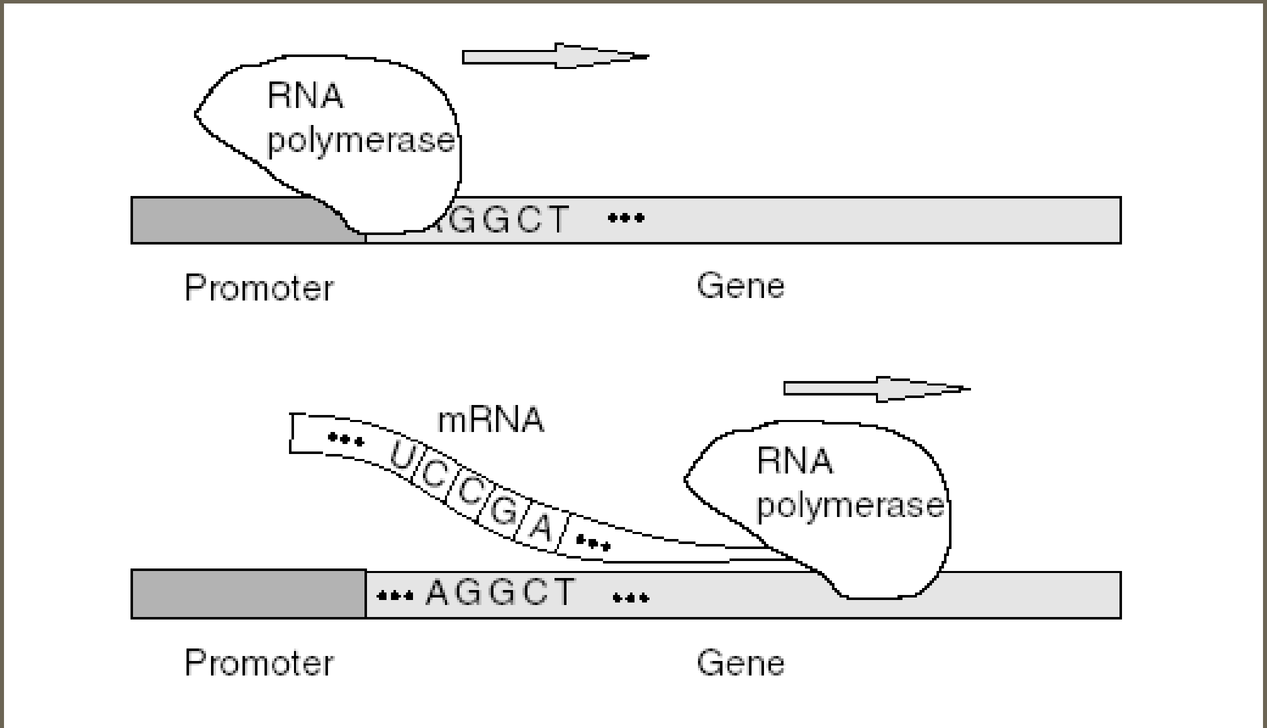
- Genes are used for making proteins, through the two steps of transcription and translation:



pp. 35-39

Transcription

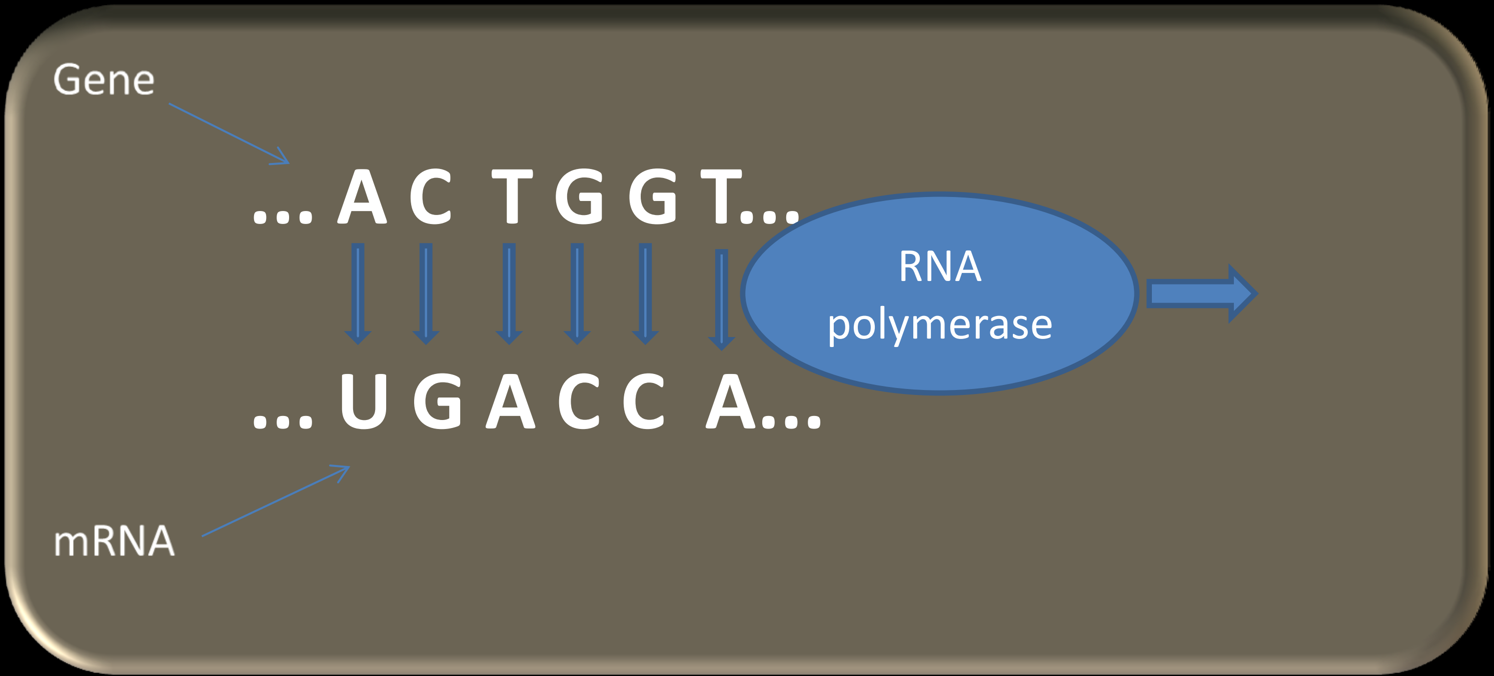
- DNA read by RNA polymerase => mRNA



pp. 35-39

Transcription

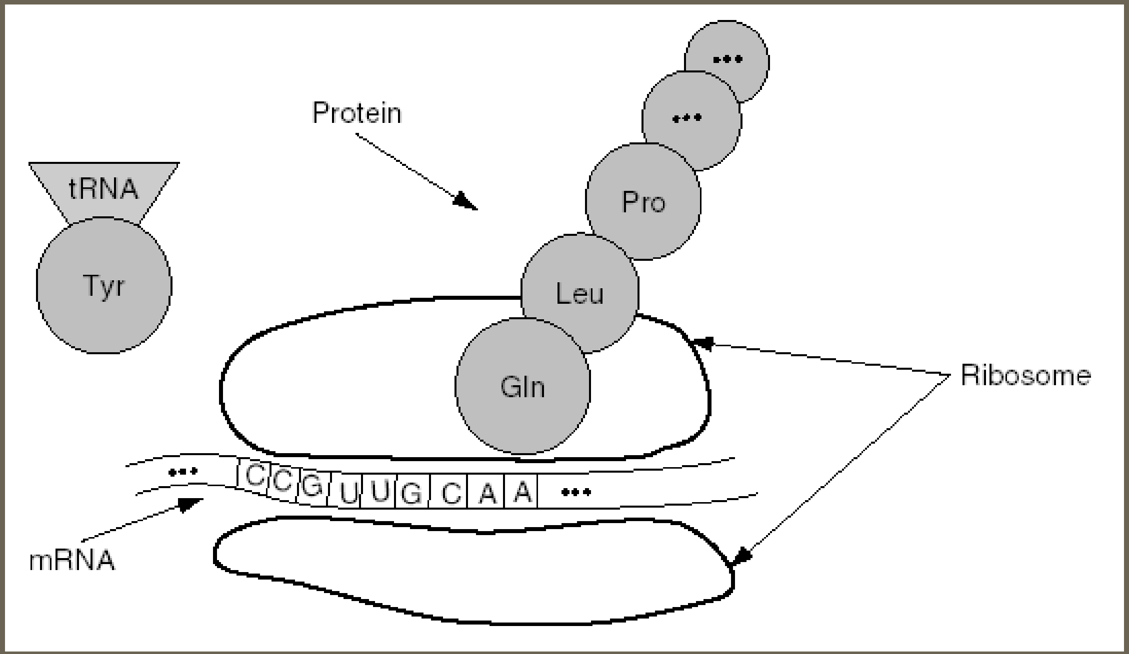
- U replaces T in mRNA



pp. 35-39

Translation

- mRNA used as a template to generate chains of amino acids (proteins):



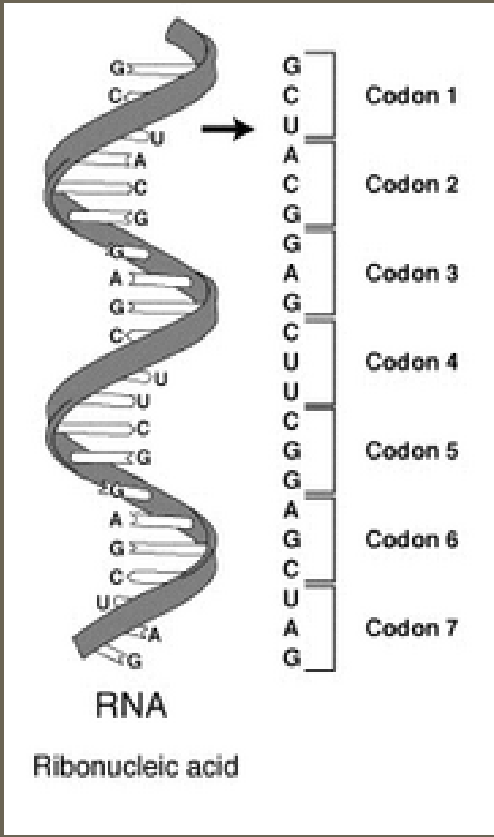
pp. 35-39

Translation

- How does the system know *which* amino acid to pick?
  - 3-letter sequences code for an amino acid.
  - There are 20 (standard) amino acids => 3 letters required to encode all in base 4 (A, C, G, U).
  - Example: The sequence CUA codes for Leucine.
  - Some sequences (e.g. AUG) code for *start* or *stop*.

pp. 35-39

Translation



Second Position

First Position	U		C		A		G		Third Position
U	UUU	Phe	UCU	Ser	UAU	Tyr	UGU	Cys	U
	UUC		UCC		UAC		UGC		C
	UUA	Leu	UCA		UAA	*Stop*	UGA	*Stop*	A
	UUG		UCG		UAG	*Stop*	UGG	Trp	G
C	CUU	Leu	CCU	Pro	CAU	His	CGU	Arg	U
	CUC		CCC		CAC		CGC		C
	CUA		CCA		CAA	Gln	CGA		A
	CUG		CCG		CAG		CGG		G
A	AUU	Ile	ACU	Thr	AAU	Asn	AGU	Ser	U
	AUC		ACC		AAC		AGC		C
	AUA		ACA		AAA	Lys	AGA	Arg	A
	AUG	Met	ACG		AAG		AGG		G
G	GUU	Val	GCU	Ala	GAU	Asp	GGU	Gly	U
	GUC		GCC		GAC		GGC		C
	GUA		GCA		GAA	Glu	GGA		A
	GUG		GCG		GAG		GGG		G

pp. 35-39

Mattias Wahde, PhD, Professor, Chalmers University of Technology
e-mail: mattias.wahde@chalmers.se, http://www.me.chalmers.se/~mwahde

Today's learning goals

- After this lecture you should be able to
  - Motivate the use of evolutionary algorithms.
  - Describe the central concepts in Darwinian evolution.
  - Describe the structure and functionality of genes (the processes of transcription and translation).
  - Summarize the main aspects of evolution.
  - Describe the basic functionality of evolutionary algorithms.
  - Exemplify some of the simplifications used in evolutionary algorithms (relative to biological evolution).

Additional concepts

- **Alleles**: The various settings (variants) of a gene
- **Genotype**: The genome with all its alleles
- **Phenotype**: The individual with all its traits, resulting from a given genotype.

pp. 35-39

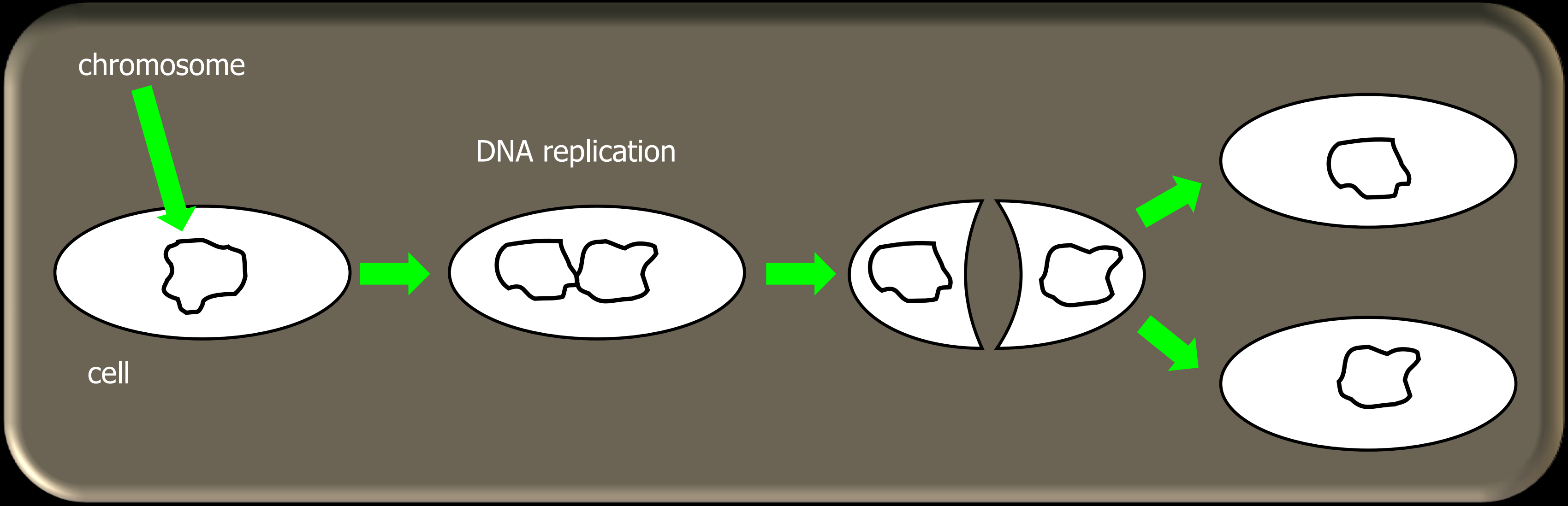
Fitness, selection, and reproduction

- The most well-adapted individuals (i.e. those with highest **fitness**) are more likely to have the opportunity (**selection**) to **reproduce** (i.e. to spread their genetic material).
- Thus, even though selection is stochastic (i.e. contains a random element) it does not occur with equal probability for individuals – instead it strongly favors fit individuals.

pp. 35-39

Reproduction

- Asexual reproduction (e.g. bacteria):



- Sexual reproduction: Combines the genetic material of two individuals.

pp. 35-39

Mattias Wahde, PhD, Professor, Chalmers University of Technology
e-mail: mattias.wahde@chalmers.se, http://www.me.chalmers.se/~mwahde

Mutations

- Mutations are small, random changes (copying errors)
- Many mutations have no effect at all, others are lethal.
- In general, even though mutations often have negative immediate effects, they give evolution new material to work (or, rather, experiment) with.

pp. 35-39

Evolution (summary)

- Evolution acts on populations of individuals (of a given species).
- Information is stored in the form of chromosomes, each consisting of many genes.
- Genes store the information needed to generate proteins (by transcription and translation).
- Well-adapted (fit) individuals spread their genetic material through reproduction.
- Mutations provide evolution with new opportunities.

pp. 35-39

Mattias Wahde, PhD, Professor, Chalmers University of Technology
e-mail: mattias.wahde@chalmers.se, http://www.me.chalmers.se/~mwahde

Today's learning goals

- After this lecture you should be able to
  - Motivate the use of evolutionary algorithms.
  - Describe the central concepts in Darwinian evolution.
  - Describe the structure and functionality of genes (the processes of transcription and translation).
  - Summarize the main aspects of evolution.
  - Describe the basic functionality of evolutionary algorithms.
  - Exemplify some of the simplifications used in evolutionary algorithms (relative to biological evolution).

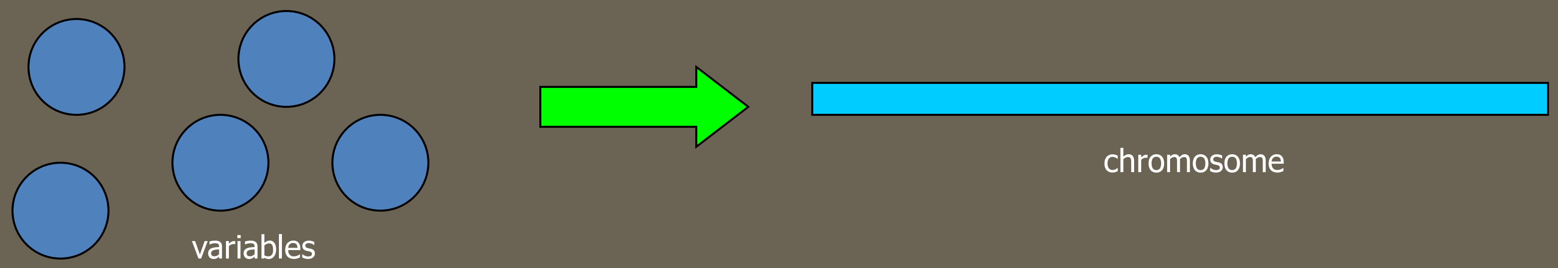
Evolutionary algorithms

- EA = evolutionary algorithm
- GA = genetic algorithm
- We will begin by describing GAs, which constitute an important special case.
- Algorithm 3.1 in the book.

pp. 40-45

Encoding

- Encoding: the variables of the problem are encoded in strings of digits known as chromosomes.



- The exact encoding procedure varies from problem to problem. Two examples (among many!):
  - Function optimization: Encode the variables x_1, x_2, x_3 etc.
  - Neural network optimization: Encode the network weights.

pp. 40-45

Mattias Wahde, PhD, Professor, Chalmers University of Technology
e-mail: mattias.wahde@chalmers.se, http://www.me.chalmers.se/~mwahde

Step 1: Initialization

- Generate N random chromosomes (first **generation**):

1	0 1 0 1 1 0 1 1 1 0 1 0 1 1 1 0 1 1 0 0 0 1 0 1 0 1 1 0 1 0 0 1 1 0 1 0 1 1 1 1 0 0 0 1 0 1 0 0 1 0 1 1 0 1
2	1 0 1 1 1 0 1 0 0 0 1 0 1 1 0 0 0 1 0 0 0 0 0 1 1 0 0 0 1 1 0 1 0 1 1 0 0 1 0 1 0 0 0 0 1 0 1 1 0 0 0 1 1 0
3	0 1 0 1 1 1 0 1 1 1 1 1 0 0 1 0 1 1 0 0 0 0 1 0 0 1 1 0 1 0 0 1 1 0 1 0 1 0 0 0 1 1 1 0 1 0 1 1 0 0 0 1 0 1
	…
N	0 1 0 1 1 0 1 1 1 0 1 0 0 0 1 1 1 0 1 1 1 0 1 1 0 1 0 0 1 0 0 1 1 0 1 0 1 1 1 1 0 0 0 1 1 1 0 0 0 0 0 0 0 1

pp. 40-45

Mattias Wahde, PhD, Professor, Chalmers University of Technology
e-mail: mattias.wahde@chalmers.se, http://www.me.chalmers.se/~mwahde

Step 2: Evaluation

- Step 2.1: Decode the chromosomes to obtain the variables.

0	1	0	1	1	0	1	0	1	1	0	0	1	1	0	1	1	0	1	0
g_1									g_{10}										

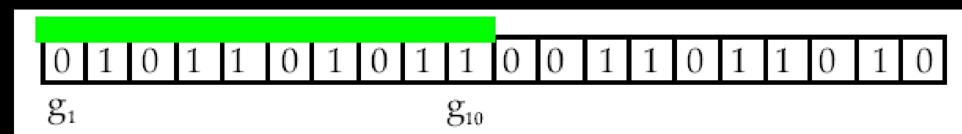
- See example 3.1 (that illustrates the decoding of one chromosome).

pp. 40-45

Mattias Wahde, PhD, Professor, Chalmers University of Technology
e-mail: mattias.wahde@chalmers.se, http://www.me.chalmers.se/~mwahde

Step 2.1: Decoding

- The first 10 bits are used to form the first variable.
- First, a value in the range [0,1[is generated:
- $x_1^{\text{tmp}} = \sum_{j=1}^{10} 2^{-j} g_j$



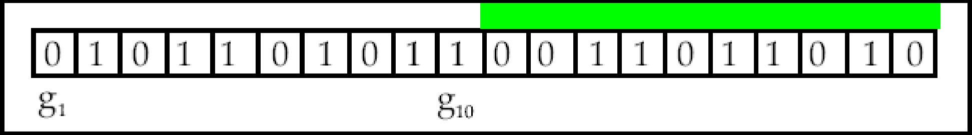
- This value is then rescaled to the required range, which in this case happened to be [-3,3]:
- $x_1 = -3 + \frac{2 * 3}{1 - 2^{-10}} x_1^{\text{tmp}}$

pp. 40-45

Mattias Wahde, PhD, Professor, Chalmers University of Technology
e-mail: mattias.wahde@chalmers.se, http://www.me.chalmers.se/~mwahde

Step 2: Decoding

- Next, x_2 is obtained in the same way, using the last 10 bits:



- $x_2^{\text{tmp}} = \sum_{j=1}^{10} 2^{-j} g_{j+10}$
- $x_2 = -3 + \frac{2 * 3}{1 - 2^{-10}} x_2^{\text{tmp}}$
- Thus, from the 20-bit chromosome, we have formed two variables, x_1 and x_2, both in the range [-3,3].

pp. 40-45

Mattias Wahde, PhD, Professor, Chalmers University of Technology
e-mail: mattias.wahde@chalmers.se, http://www.me.chalmers.se/~mwahde

CHALMERS

Step 2: Evaluation

- Step 2.2 Evaluate the individual, and assign a fitness score.
- Requirement: Fitness measure. Examples:
  - Function maximization: Use simply $f(x)$.
  - Function minimization: Use (e.g.) $1/f(x)$.
- See example 3.2.
- Step 2.3: Repeat Steps 2.1 and 2.2 for all individuals => a set of fitness scores is obtained, one for each indivdual.

pp. 40-45

Step 3: Form the next generation

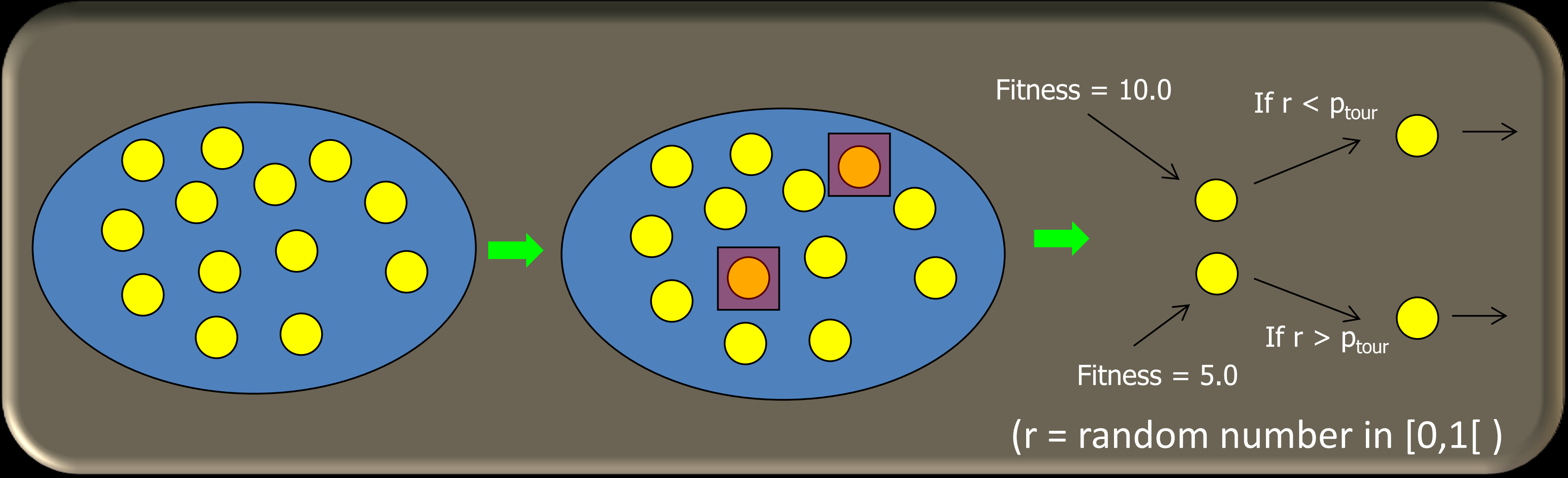
- Step 3.1: Select two individuals (one at a time), such that individuals with high fitness have larger probability of selection.
- Various methods exist, e.g. Tournament selection, Roulette-wheel selection etc.
- (These methods will be studied in detail in the next lecture!)
- Selection is carried out with replacement, i.e. individuals can be selected more than once.

pp. 40-45

Mattias Wahde, PhD, Professor, Chalmers University of Technology
e-mail: mattias.wahde@chalmers.se, http://www.me.chalmers.se/~mwahde

Example: Tournament selection

- Procedure for selection of *one* individual:
    1. Pick two individuals randomly from the population (equal probability for all individuals in the population)
    2. With probability p_{tour} (usually around 0.7-0.8) pick the better of the two individuals (higher fitness). Otherwise, pick the worse of the two.

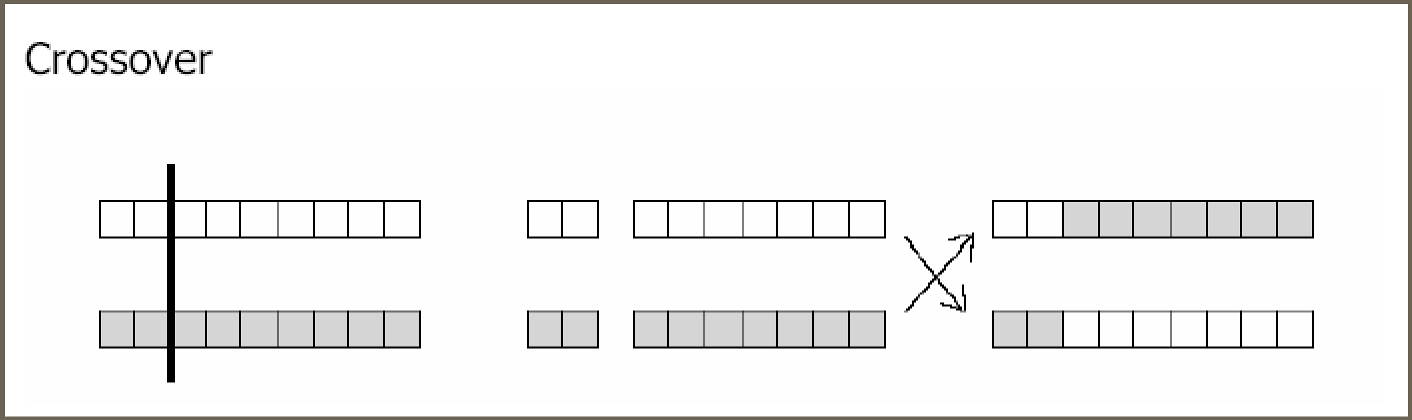


Mattias Wahde, PhD, Professor, Chalmers University of Technology
e-mail: mattias.wahde@chalmers.se, http://www.me.chalmers.se/~mwahde

CHALMERS

Step 3: Form the next generation

- Step 3.2: Generate two new chromosomes by crossing the chromosomes of the two selected individuals:

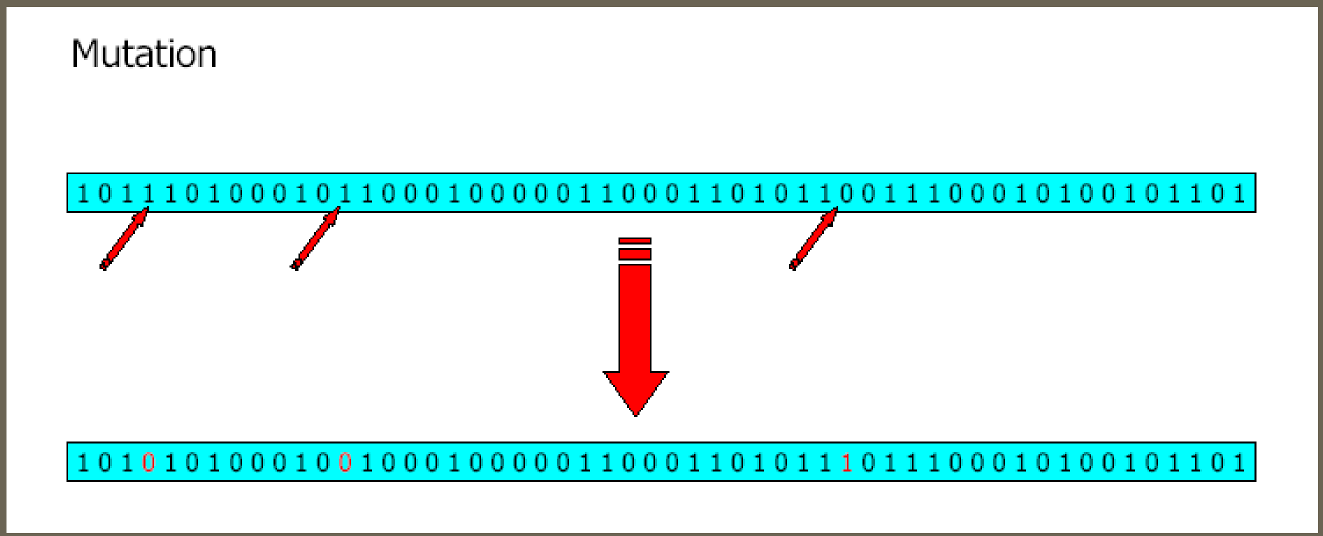


pp. 40-45

Mattias Wahde, PhD, Professor, Chalmers University of Technology
e-mail: mattias.wahde@chalmers.se, http://www.me.chalmers.se/~mwahde

Step 3: Form the next generation

- Step 3.3: Mutate the two new chromosomes:



pp. 40-45

Mattias Wahde, PhD, Professor, Chalmers University of Technology
e-mail: mattias.wahde@chalmers.se, http://www.me.chalmers.se/~mwahde

Step 3: Form the next generation

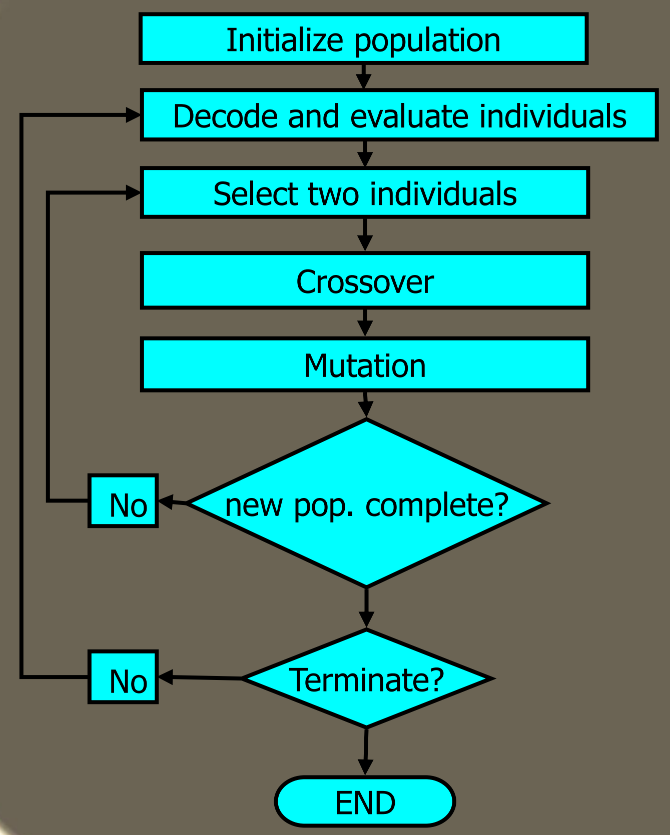
- Step 3.4: Repeat the entire procedure (Steps 3.1 – 3.3) $N/2$ times, to generate N new individuals. That is, do $N/2$ repetitions of...
  - ...selection of two individuals (for example by doing tournament selection twice),
  - ...crossover, and
  - ...mutation of the resulting pair of chromosomes
- Then replace the old set of chromosomes by the new ones.

pp. 40-45

Mattias Wahde, PhD, Professor, Chalmers University of Technology
e-mail: mattias.wahde@chalmers.se, http://www.me.chalmers.se/~mwahde

Step 4: Repeat!

- The process is repeated (from Step 2) until a satisfactory solution has been found.



pp. 40-45

Mattias Wahde, PhD, Professor, Chalmers University of Technology
e-mail: mattias.wahde@chalmers.se, http://www.me.chalmers.se/~mwahde

Example (Fig. 3.7)

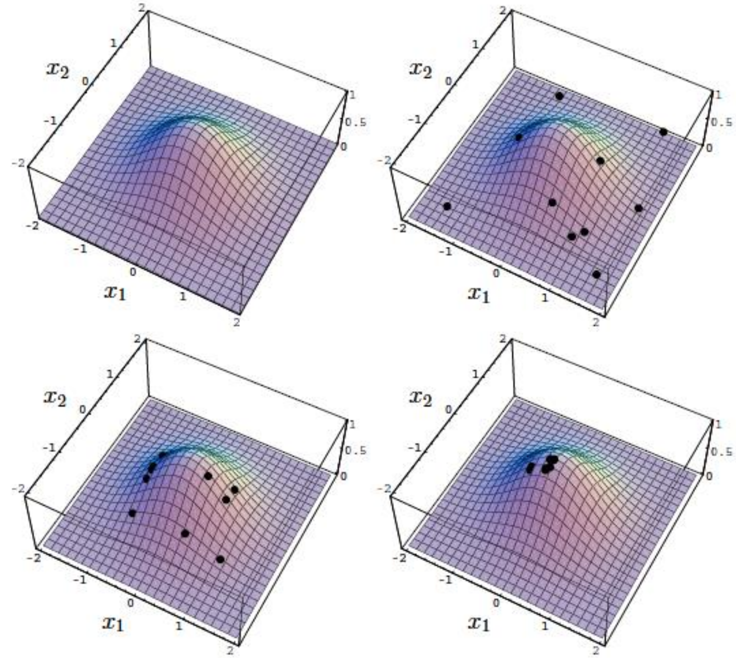


Figure 3.7: *The progress of a GA searching for the maximum of the function* $f(x_1, x_2) = e^{-x_1^2 - x_2^2}$. *The upper right panel shows the initial population, whereas the lower left and lower right panels show the population after two and 25 generations, respectively.*

Today's learning goals

- After this lecture you should be able to
  - Motivate the use of evolutionary algorithms.
  - Describe the central concepts in Darwinian evolution.
  - Describe the structure and functionality of genes (the processes of transcription and translation).
  - Summarize the main aspects of evolution.
  - Describe the basic functionality of evolutionary algorithms.
  - Exemplify some of the simplifications used in evolutionary algorithms (relative to biological evolution).

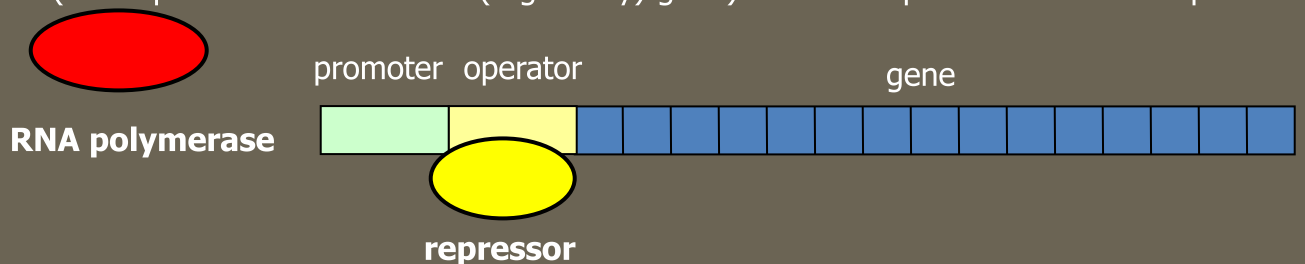
Evolution vs. evolutionary algorithms

- Evolutionary algorithms (EAs) are strongly simplified compared to biological evolution!
  - In EAs, chromosomes are used as lookup tables, i.e. there are no gene interactions etc.
  - In EAs, multicellularity (difference between cells etc.) is rarely considered. In biological systems, cell share the same genetic material, but develop differently (skin, bone, muscle etc.)
  - In EAs, there is no gene regulation. In biological systems, (many) genes are active, to varying degrees, during the life time of the individual.

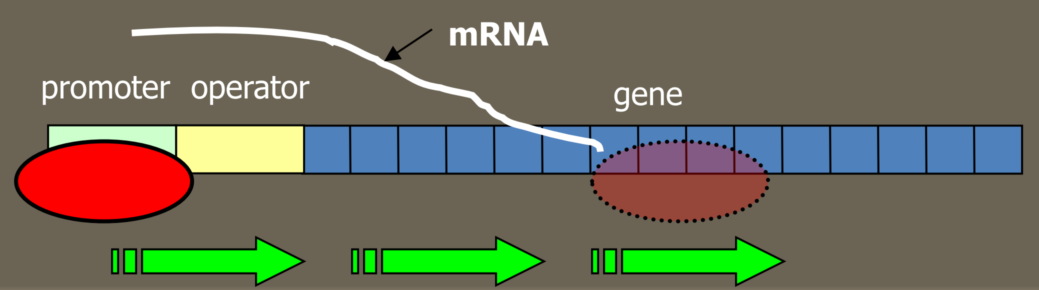
Gene regulation

- Regulatory genes: (transcription factors) genes that regulate the expression of other genes. Example:

Repressor protein (= the product of some other (regulatory) gene) bound to operator site: transcription is prevented



Repressor *not* bound to operator: the RNA polymerase can reach the promoter and proceed with transcription:



Mattias Wahde, PhD, Professor, Chalmers University of Technology
e-mail: mattias.wahde@chalmers.se, http://www.me.chalmers.se/~mwahde

Today's learning goals

- After this lecture you should be able to
  - Motivate the use of evolutionary algorithms. ✓
  - Describe the central concepts in Darwinian evolution. ✓
  - Describe the structure and functionality of genes (the processes of transcription and translation). ✓
  - Summarize the main aspects of evolution. ✓
  - Describe the basic functionality of evolutionary algorithms. ✓
  - Exemplify some of the simplifications used in evolutionary algorithms (relative to biological evolution). ✓

Home problem 1

- Consists of three parts.
  - Problems 1.1 and 1.3 are mandatory,
  - Problem 1.2 is voluntary (but needed for higher grades).
- Use Matlab when solving the home problems.
- Maximum total score: 10p.
- Should be solved (and handed in) *independently by each student*.
- Strict deadline: 20200922, 23.59.59.

Home problem 1

- Note: Read the instructions carefully (regarding the report, coding standard etc.)
- Before submitting:
  - What, ***exactly***, is being asked in the problem(s)? Have you ***answered*** those questions?
  - Have you written and included a report in the required format?
  - Have you combined your programs *and* your report into ONE compressed file, in an allowed format (.zip or .7z)?
- Read the checklist (available on the web page, under the module called Miscellaneous).

Problem 1.1

- The penalty method:

In this problem, we shall use the penalty method (see pp. 30-33 in the course book) to find the minimum of the function

$$f(x_1, x_2) = (x_1 - 1)^2 + 2(x_2 - 2)^2, \tag{1}$$

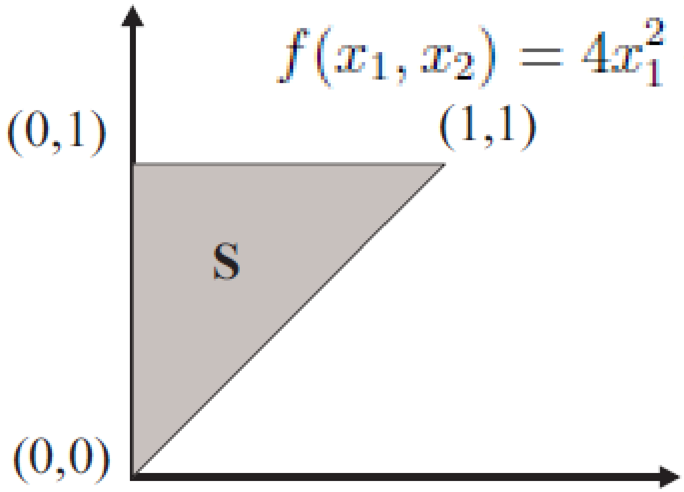
subject to the constraint

$$g(x_1, x_2) = x_1^2 + x_2^2 - 1 \leq 0. \tag{2}$$

Problem 1.2

- (Classical) optimization under constraints

$$f(x_1, x_2) = 4x_1^2 - x_1x_2 + 4x_2^2 - 6x_2,$$



Mattias Wahde, PhD, Professor, Chalmers University of Technology
e-mail: mattias.wahde@chalmers.se, http://www.me.chalmers.se/~mwahde

Problem 1.3

- Function optimization using genetic algorithms.
- Write a (*well-structured*) program for finding the minimum of the function:

$$
\begin{aligned}
g(x_1, x_2) \;=\; & \left(1 + (x_1 + x_2 + 1)^2 (19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)\right) \times \\
& \left(30 + (2x_1 - 3x_2)^2 (18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)\right)
\end{aligned}
$$

Mattias Wahde, PhD, Professor, Chalmers University of Technology
e-mail: mattias.wahde@chalmers.se, http://www.me.chalmers.se/~mwahde

Matlab introduction for SOAs

- This afternoon and evening. The document will be posted at 13.00, I will be online (Zoom) between 18.00 and 20.00.
- Important to attend this session!
- You will get a document with Matlab code describing the implementation of a genetic algorithm (GA).
- The GA will be applied in a function optimization task.