Evolutionary Algorithms

Evolutionary Strategies, Behavioural Simulation

Prof. Dr. Rudolf Kruse **Pascal Held**
{kruse,pheld}@iws.cs.uni-magdeburg.de
Otto-von-Guericke-Universität Magdeburg
Fakultät für Informatik
Institut für Wissens- und Sprachverarbeitung

Outline

1. Evolutionary Strategies
- Selection
- Variance adaptation
- Covariance and Correlation
- Crossover/Recombination
- Plus- versus Comma-Strategies

2. Evolutionary Algorithms for Behavioral Simulation

Evolutionary Strategies (ES)

- here: focus on **numerical optimization**
  given: Function $f : \mathbb{R}^n \to \mathbb{R}$
  wanted: minimum or maximum of f
  Chromosomes: **Vectors of real numbers**
  Mutation: adding a normal-distributed **random vector** $\boldsymbol{r}$
  each element $r_i \in \boldsymbol{r}$ is a sample of a normal-distributed random number with
  - expected value 0 (independent of element index i) and
  - variance σ_i^2 resp. standard deviation σ_i
  - σ_i^2 (in)dependent of element index i and generation t
- **Crossover**: here, it's not used.

Selection

Strict elite principle:

- only the best individuals enter the next generation
- Definition:

  μ – number of individuals in the parent generation
  λ – number of offspring individuals that were created by mutation

Two essential selection strategies:

- **+-strategy** (plus strategy, $(\mu + \lambda)$-strategy)
  selection works on $(\mu + \lambda)$ individuals and the best μ chromsomes are selected for the next generation
  (it often holds $\lambda < \mu$)
- **,-strategy** (comma strategy, (μ, λ)-strategy)
  generates offspring of size $\lambda > \mu$ where the best μ chromosomes are selected
  (chromosomes of parent generation are lost)

Selection

Example: special case f (1+1)-strategy

„Initial population“: $\boldsymbol{x}_0$ (randomly generated vector of real numbers)

creating the next generation:

1. generate real random vector $\boldsymbol{r}_t$ and compute $\boldsymbol{x}_t^* = \boldsymbol{x}_t + \boldsymbol{r}_t$
2. use

$$\boldsymbol{x}_{t+1} = \begin{cases} \boldsymbol{x}_t^*, & \text{falls } f(\boldsymbol{x}_t^*) \geq f(\boldsymbol{x}), \\ \boldsymbol{x}_t, & \text{sonst.} \end{cases}$$

generate further generations until termination criterion is fulfilled

- relates to **hill climbing**

⇒ general +-strategy = parallel hill climbing, which is performed simultaneously on several parts of Ω (whereas the most promising μ paths are pursued)

Evolutionary strategies

Does not only optimizes organisms but complete mechanisms of evolution: reproduction- and mortality rate, life span, susceptibility to mutation, mutation step width, speed of evolution etc.

adaptation of random vector's variance (mutation step width)

- small variance ⇒ small changes of chromosomes ⇒ local search (exploitation)
- high variance ⇒ big changes of chromosomes ⇒ global search (exploration)

further approaches to adapt parameters:

- Choice of the number of genes which are changed (Vektorelemente)
- Choice of λ representing the amount of offspring

Global-Variance-Adaption

Chromosome independent variance

- Choose σ in such a way that the mean convergence rate is (approximately) optimized

  [Rechenberg, 1973] approach: determine an optimal σ for
  - $f_1(x_1, \ldots, x_n) = a + bx_1$ and
  - $f_2(x_1, \ldots, x_n) = \sum_{i=1}^{n} x_i^2$,

  by determining the probabilities for a successful (that is, improving) mutation.
- Results of this empirical study
  - for f_1: $p_1 \approx 0.184$ und
  - for f_2: $p_2 \approx 0.270$

  $\frac{1}{5}$-success rule
  - heuristically inferred
  - under the plus strategy the mutation step size is appropriate if approximately $\frac{1}{5}$ of the offspring are better than the parents

Global-Variance-Adaption

Adaptation of the variance σ^2 based on the $\frac{1}{5}$-success rule:

- if more than $\frac{1}{5}$ of the children are better than the parents, the variance should be increased

$$\sigma' = \sigma \cdot \alpha, \qquad \alpha > 1$$

- if less than $\frac{1}{5}$ of the children are better than the parents, the variance should be reduced:

$$\sigma' = \sigma / \alpha$$

- for larger populations the $\frac{1}{5}$-success rule is sometimes too optimistic
- in analogy to **simulated annealing**: one may define a function that increases the threshold over time

Global-Variance-Adaption

Algorithm 1 ADAPTIVE-ADAPTION

Input: Standard deviation σ, success rate p_s, threshold $\theta = \frac{1}{5}$, modification factor $\alpha > 1$

Output: adapted standard deviation σ

```
if $p_s > \theta$ {
    return $\alpha \cdot \sigma$
}
if $p_s < \theta$ {
    return $\sigma / \alpha$
}
return $\sigma$
```

Algorithm 2 ES-Adaptiv

```
Input: function $F$, population size $\mu$, number of offspring $\lambda$, modification frequency $k$
$t \leftarrow 0$
$\sigma \leftarrow$ value for the initial step size
$s \leftarrow 0$
$P(t) \leftarrow$ create a population with $\mu$ individuals
evaluate $P(t)$ with $F$
while termination criterion is not fulfilled {
    $P' \leftarrow \emptyset$                                /* for plus strategy $P' \leftarrow P(t)$ */
    for $i = 1, \dots, \lambda$ {
        $A \leftarrow$ select random parent uniformly from $P(t)$
        $B \leftarrow$ GAUSSIAN-MUTATION($A$) with $\sigma$
        evaluate $B$ with $F$
        if $B.F \succ A.F$ {
            $s \leftarrow s + 1$
        }
        $P' \leftarrow P' \cup \{B\}$
    }
    $t \leftarrow t + 1$
    $P(t) \leftarrow$ Selection in $P'$ with BEST-SELECTION
    if mod $(t, k) = 0$ {
        $\sigma \leftarrow$ ADAPTIVE-ADAPTION($\sigma, \frac{s}{k \cdot \lambda}$)
        $s \leftarrow 0$
    }
}
return best individual in $P(t)$
```

Local Variance Adaptation

chromosome-specific variances

Variance/Standard deviation is used as additional genetic information:

- a variance for all vector entries
- an individual variance for every entry of the vector (double vector length)

Note: additional entries of the vector for variances have *no direct influence* on the fitness of the chromosome

Expectation: chromosomes with „bad" variances, that is

- too small: chromosomes are growing up too slowly or
- too big: chromosomes clearly diverge from its parents

generate comparatively more „bad" offspring

its genes (and so the variances) become extinct

Local Variance Adaptation

element-specific mutation step widths (standard deviation) are mutated with the following scheme:

$$\sigma_i' = \sigma_i \cdot \exp(r_1 \cdot N(0,1) + r_2 \cdot N_i(0,1)).$$

$N(0,1)$: one normal distributed number per chromosome

$N_i(0,1)$: one normal distributed number per gene

recommended values for r_1, r_2 [Bäck and Schwefel, 1993]

$$r_1 = \frac{1}{\sqrt{2n}}, \qquad r_2 = \frac{1}{\sqrt{2\sqrt{n}}},$$

where n is number of vector entries, or [Nissen, 1997]

$$r_1 = 0.1, \qquad r_2 = 0.2$$

often: lower bound for mutation step widths

Algorithm 3 ES-Selbstadaptiv

```
Input: function $F$, population size $\mu$, number of offspring $\lambda$
$t \leftarrow 0$
$P(t) \leftarrow$ create population with $\mu$ individuals
evaluate $P(t)$ with $F$
while termination criterion not fulfilled {
    $P' \leftarrow \emptyset$                    /* for plus selection $P' \leftarrow P(t)$ */
    for $i = 1, \ldots, \lambda$ {
        $A \leftarrow$ select parent uniformly random from $P(t)$
        $B \leftarrow$ SELF-ADAPTIVE-GAUSSIAN-MUTATION($A$)
        $P' \leftarrow P' \cup \{B\}$
    }
    evaluate $P'$ with $F$
    $t \leftarrow t + 1$
    $P(t) \leftarrow$ select best $\mu$ individuals from $P'$
}
return best individual in $P(t)$
```

Extension of the local variance adaptation

- standard form of the local variance adaptation: variances of the different vector entries are independent of each other
  (formal: Covariance matrix is a diagonal matrix)
- for variants of chromosomes favored in certain directions: adaptation with single variance if and only if orthogonal directions

  Example: create variants of chromosomes with two genes favored in the direction of the principle diagonal, that is $(1, 1)$

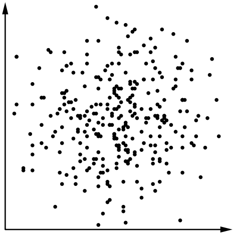
  not describable with single variances

  Solution: use covarianz matrix with a high covariance, e.g.

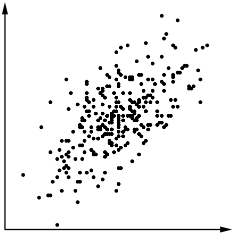
$$\Sigma = \begin{pmatrix} 1 & 0.9 \\ 0.9 & 1 \end{pmatrix}$$

Covariance and Correlation

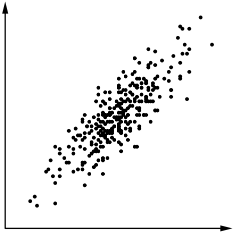
no
correlation



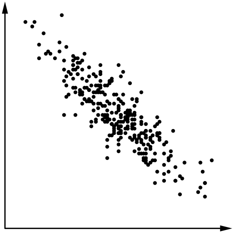
weak
positive
correlation



strong
positive
correlation



strong
negative
correlation



Cholesky-Decomposition

S symmetric positive definite matrix (d.h. covariance matrix)

- symmetric: $\forall 1 \leq i, j \leq m : s_{ij} = s_{ji}$
- positive definite: for all m-dim. Vectors $\boldsymbol{v} \neq \mathbf{0}$, it holds $\boldsymbol{v}^T \mathbf{S} \boldsymbol{v} > 0$

Cholesky-Decomposition: Computation of the „Square-Root“ of **S**

Compute left lower triangle matrix **L** so that $\mathbf{L}\mathbf{L}^T = \mathbf{S}$
($\mathbf{L}^T$ is transpose of the matrix **L**)

$$l_{ii} = \left(s_{ii} - \sum_{k=1}^{i-1} l_{ik}^2 \right)^{\frac{1}{2}}$$

$$l_{ji} = \frac{1}{l_{ii}} \left(s_{ij} - \sum_{k=1}^{i-1} l_{ik} l_{jk} \right), \qquad j = i+1, i+2, \ldots, m$$

Cholesky-Decomposition

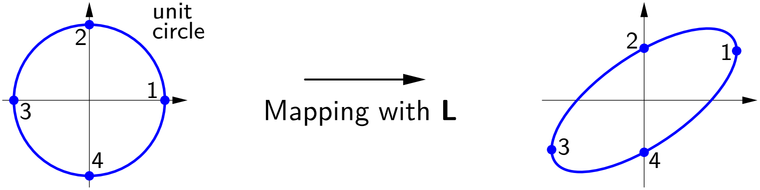
Special case: two dimensions

Covariance matrix

$$\Sigma = \begin{pmatrix} \sigma_x^2 & \sigma_{xy} \\ \sigma_{xy} & \sigma_y^2 \end{pmatrix}$$

Cholesky-Decomposition

$$\mathbf{L} = \begin{pmatrix} \sigma_x & 0 \\ \dfrac{\sigma_{xy}}{\sigma_x} & \dfrac{1}{\sigma_x}\sqrt{\sigma_x^2\sigma_y^2 - \sigma_{xy}^2} \end{pmatrix}$$



Eigenvalue Decomposition

also yields to **analogue of the standard deviation**

more expensive to compute than Cholesky-Decomposition

S symmetric positive definite matrix (d.h. covariance matrix)

$$\mathbf{S} = \mathbf{R}\,\mathrm{diag}(\lambda_1, \dots, \lambda_m)\,\mathbf{R}^{-1},$$

whereas λ_j, $j = 1, \dots, m$ are Eigenvalues of **S** and columns **R** are (normalised) Eigenvectors of **S**

- Eigenvalues λ_j, $j = 1, \dots, m$ of **S** are positive and Eigenvectors of **S** are orthonormal ($\Rightarrow \mathbf{R}^{-1} = \mathbf{R}^T$)

$\mathbf{S} = \mathbf{T}\mathbf{T}^T$ with

$$\mathbf{T} = \mathbf{R}\,\mathrm{diag}\left(\sqrt{\lambda_1}, \dots, \sqrt{\lambda_m}\right)$$

Eigenvalue Decomposition

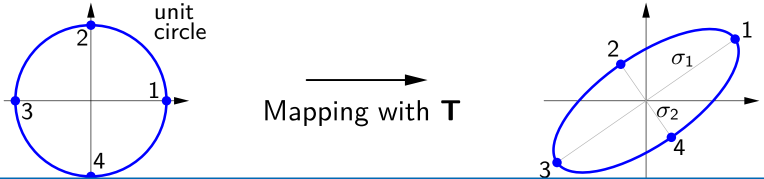
Special case: Two dimensions

Covariance matrix

$$\Sigma = \begin{pmatrix} \sigma_x^2 & \sigma_{xy} \\ \sigma_{xy} & \sigma_y^2 \end{pmatrix}$$

Eigenvalue Decomposition
$(s = \sin\phi,\ c = \cos\phi,\ \phi = \frac{1}{2} \arctan \frac{\sigma_{xy}}{\sigma_y^2 - \sigma_x^2})$

$$\mathbf{T} = \begin{pmatrix} c & s \\ -s & c \end{pmatrix} \begin{pmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{pmatrix}, \qquad \begin{aligned} \sigma_1 &= \sqrt{c^2\sigma_x^2 + s^2\sigma_y^2 - 2sc\sigma_{xy}}, \\ \sigma_2 &= \sqrt{s^2\sigma_x^2 + c^2\sigma_y^2 + 2sc\sigma_{xy}}. \end{aligned}$$



Variance adaptation

in general: describe correlated mutation with n variances and $\frac{n(n-1)}{2}$ rotation angles

use covariance matrix

$$\Sigma = \left(\prod_{i=1}^{n}\prod_{k=i+1}^{n} R_{ik}(\varphi_{ik})\right) \mathrm{diag}(\sigma_1^2, \ldots, \sigma_n^2) \left(\prod_{i=1}^{n}\prod_{k=i+1}^{n} R_{ik}(\varphi_{ik})\right)^{-1}$$

wobei

$$R_{ik}(\varphi_{ik}) = \begin{pmatrix}
1 &&&&&&&&&& \\
& \ddots &&&&&&&&& \\
&& 1 &&&&&&&& \\
&&& \cos\varphi &&&& \sin\varphi &&& \\
&&&& 1 &&&&&& \\
&&&&& \ddots &&&&& \\
&&&&&& 1 &&&& \\
&&& -\sin\varphi &&&& \cos\varphi &&& \\
&&&&&&&& 1 && \\
&&&&&&&&& \ddots & \\
&&&&&&&&&& 1
\end{pmatrix}$$

Variance adaptation

Covariance matrix Σ is generally chromsome-specific (then, chromosome has $n + \frac{n(n+1)}{2}$ genes)

Mutation of covariances on rotation angles and not directly on the matrix entries:

$$\varphi'_{ik} = \varphi_{ik} + r \cdot N(0,1)$$

with $r \approx 0.0873$ ($\approx 5^\circ$)

$N(0,1)$ in every step a new normal distributed number is generated

disadvantages of the correlated mutation:

- notable more parameters have to be adapted
- variances and rotation angles have no direct influence on the fitness function; its adaptation is performed rather casually

Crossover/Recombination

random selection of components of the parents:

$$\begin{matrix}(\mathbf{x_1},\ x_2,\ x_3,\ \ldots,\ \mathbf{x_{n-1}},\ x_n)\\(y_1,\ \mathbf{y_2},\ \mathbf{y_3},\ \ldots,\ y_{n-1},\ \mathbf{y_n})\end{matrix} \quad\Rightarrow\quad (x_1, y_2, y_3, \ldots, x_{n-1}, y_n)$$

corresponds to **uniform crossover**
(basically, 1-, 2- or *n*-point crossover can be used, too)
Averaging (blending, intermediäry recombination):

$$\begin{matrix}(x_1,\ \ldots,\ x_n)\\(y_1,\ \ldots,\ y_n)\end{matrix} \quad\Rightarrow\quad \frac{1}{2}(x_1 + y_1, \ldots, x_n + y_n)$$

Attention: when using blending method, danger of **Jenkins Nightmare** is on the spot

- total disappearance of any diversity in a population
- benefited by averaging since genes head on mid values

Plus- versus Comma-Strategies

Advantage of the $+$-strategy:

- due to strict elite principle: only improvements

Disadvantages:

- risk of getting stuck in local optima
- for $(\mu + \lambda)$-strategy with $\frac{\mu}{\lambda} \geq$ „best probability for a successful mutation“ ($\approx \frac{1}{5}$) Chromosomes have a selection benefit, which keep its variance σ^2 as low as possible since not enough mutations are performed to get a substantial improvement („Beinahe-Stagnation“)
  common choice of the fraction of μ and λ approx. 1:7

if no improvement over further generations is recognizable, it is temporary helpful to switch off to the ,-strategy to overcome local minima (increases diversity in population)

Outline

1. Evolutionary Strategies

2. **Evolutionary Algorithms for Behavioral Simulation**
    - The prisoner's dilemma
    - A Genetic Algorithm Approach
    - Extensions

Evolutionary Algorithms for Behavioral Simulation

before: EAs are used to solve (numerical or discrete) optimization problems

now: using EAs to simulate behaviour(population dynamics) and to find behavioural strategies

basis: game theory

- used to analyze social and economic situations
- rationale is to model agents and their actions as game moves in a formally specified framework
- most important theoretical foundation of economics

General Approach:

- encoding the moves of an agent in an chromosome
- let agents interact with each other and evaluate their success
- agent reproduce or die due to their achieved success

The prisoner's dilemma

best-known and most thoroughly studied problem of the game theory is the so-called **prisoner's dilemma**(PD)

- 2 people robbed a bank and were arrested
- but the available circumstantial evidence is not sufficient for a conviction because of the bank robbery
- There is, however, sufficient evidence for a conviction because of a lesser criminal offense (say, illegal possession of firearms) prosecutor offers both prisoners to become a key witness
  - if one of them confesses to the bank robbery, he/she is exempted from punishment
  - the other prisoner will be punished with the full force of the law (10 years imprisonment)
  - Problem: both prisoners are offered this possibility and thus both may be tempted to confess
    Since they both pleaded guilty, though, they receive a mitigated sentence, meaning that both of them have to spend 5 years in prison

The prisoner's dilemma

Analyze of the prisoner's dilemma with **payoff matrix**:

B A	cooperate	defect
cooperate	**−1** **−1**	**0** **−10**
defect	**−10** **0**	**−5** **−5**

Keeping silent is favorable for both

But: a double confession is the so-called **Nash equilibrium**: No agent can improve its payoff by changing its action (each payoff matrix has at least one Nash equilibrium [Nash, 1950])

Gen. payoff matrix of the prisoner's dilemma

B A	cooperate	defect
cooperate	R R	T S
defect	S T	P P

R: Reward for mutual cooperation
P: Punishment for mutual defection
T: Temptation to defect
S: Sucker's payoff

- exact values for **R**, **P**, **T** and **S** are not important
- but it definitely holds $\mathbf{T} > \mathbf{R} > \mathbf{P} > \mathbf{S}$ and $2\,\mathbf{R} > \mathbf{T} + \mathbf{S}$
  2. condition not fulfilled ⇒ alternately exploitation should be preferred

The prisoner’s dilemma

Many situations in everyday life are analogous to the PD

but: although double defect = Nash equilibrium, other cooperating actions

Question (accord. to [Axelrod, 1980]):
Under what conditions cooperation emerges in a world of egoists without any central authority?

Answer from [Hobbes, 1651] (Leviathan):

- **Under no conditions whatsoever!** Before governmental order and thus a directing central authority existed, the state of nature was dominated by egoistic individuals that competed against each other in such a reckless way that life was „solitary, poor, nasty, brutish, and short“
- **however:** on an international level, there is *de facto* no central authority but there is a cooperation of countries, though.

The prisoner's dilemma

Approach from [Axelrod, 1980]: **iterated prisoner's dilemma**. (dilemma is performed several times consecutively where all previous actions of the agents are known)

Idea:

- is the dilemma performed *once*, it is favorable to choose the nash equilibrium
- is the dilemma performed *several times*, one agent can react on the incoorperative behaviour
  (possibility of *retaliation*)

Questions:

1. **Is cooperation created in the iterated prisoner's dilemma?**
2. **What is the best strategy in the iterated prisoner's dilemma?**

The prisoner’s dilemma

[Axelrod, 1980] specified the shown **payoff matrix**:

B A	cooperate	defect
cooperate	3 3	5 0
defect	0 5	1 1

(smallest non-negative integer numbers that satisfy the two conditions)

- Axelrod invited scientists from diverse disciplines (psychology, social and political sciences, economics, mathematics) to encode what they believed to be an optimal strategy for the iterated prisoner’s dilemma with this payoff matrix
- programs were to have access to all games already played

Turniere

With this framework, Axelrod conducted two tournaments:

1. **Tournament**:
    - 14 Programs and one random player (Fortran)
    - competed against each other in a round-robin tournament with 200 matches per pairing
    - Winner: A. Rapoport with Tit-for-Tat

program code of all participants of this tournament together with the payoff results was published

invited to a second tournament

idea: analyzing the results of the first tournament

2. **Tournament**:
    - 62 programs and one random player participated (Fortran und Basic)
    - same tournament conditions
    - Winner: A. Rapoport with Tit-for-Tat

Tit-for-Tat

Strategy of **Tit-for-Tat** is *very* simple:

- cooperate in the first game (action: C)
- react in all following games with the move of the opponent's previous played game

Please note: tit for tat does not win generally against any other strategy

- if there are agents in the population with whom it can cooperate, it can gain an overall advantage
- **Problem** of Tit-for-Tat: may react **inadequately to mistakes** — if two instances of Tit-for-Tat play against each other and one instance „accidentally“ plays defect, this results i mutual retaliations

A reasonable alternative is **Tit-for-Two-Tat**:
strategy starts retaliating only after having been exploited twice

A Genetic Algorithm Approach

Encoding of the strategies: [Axelrod, 1987]

consider all possible sequences of three consecutive games (6 moves: $2^6 = 64$ possible sequences)

store what move should be played in the next game (C – cooperate, D – defect, in 1 Bit):

		1. game	2. game	3. game	
1. Bit:	response to	(C,C),	(C,C),	(C,C):	C
2. Bit:	response to	(C,C),	(C,C),	(C,D):	D
3. Bit:	response to	(C,C),	(C,C),	(D,C):	C
⋮		⋮		⋮	⋮
64. Bit:	response to	(D,D),	(D,D),	(D,D):	D

(1. and 2. element of each pair: own resp. opposing move)

6 Bit that encode the course of the game „before“ the first move

each chromosome has 70 binary genes (either C or D)

Genetic Algorithm: procedure

initial population is created by randomly sampling bit sequences of length 70

in current population: choose pairs of individuals randomly

pairs play the prisoner's dilemma 200 times

on the first 3 games: use the stored history of the beginning of the games to determine the moves
(missing/too short history is filled)

each individual plays against the same number of opponents
(due to limitation on computing power – 1987! – no complete tournament)

Auswahl von Individuen für nächste Generation:
over-average result ($x \geq \mu + \sigma$): 2 children
average result ($\mu - \sigma < x < \mu + \sigma$): 1 child
below-average result ($\mu - \sigma \geq x$): no offspring

genetic operators: Bit-Mutation, one-point crossover

Genetic Algorithm: Result

sich ergebende Strategien sind **Tit-for-Tat** sehr ähnlich

[Axelrod, 1987] identified the following general patterns:

Don't rock the boat: Cooperate after three times cooperate
(C,C), (C,C), (C,C) → C

Be provokable: Play defect after a sudden defect of the opponent
(C,C), (C,C), (C,D) → D

Accept an apology: Cooperate after mutual exploitation
(C,C), (C,D), (D,C) → C

Forget: (Do not be resentful:) Cooperate after cooperation has been restored after one defect (also without retaliation)
(C,C), (C,D), (C,C) → C

Accept a rut:
Play defect after three times defect of the opponent
(D,D), (D,D), (D,D) → D

The prisoner's dilemma: Extensions

- in the real world: consequences of actions are not always perfectly observable
- it may not always be perfectly clear whether last move of opponent was actually a defect
- more than two agents are involved: Multiple-Agent-Prisoner's dilemma

- longer match histories may be considered
- using a random component on the choice of the move: Probability to choose C or D instead of a clear determined move
- Description of Moore machines or even general programs that are evolved with the principles of genetic programming

Further reading I

Axelrod, R. (1980).
More effective choice in the prisoner's dilemma.
Journal of Conflict Resolution, 24:379–403.

Axelrod, R. (1987).
The evolution of strategies in the iterated prisoner's dilemma.
In Davis, L., editor, *Genetic Algorithms and Simulated Annealing*, pages 32–41. Morgan Kaufman, Los Altos, CA, USA.

Bäck, T. and Schwefel, H. (1993).
An overview of evolutionary algorithms for parameter optimization.
Evolutionary Computation, 1(1):1–23.

Hobbes, T. (1651).
Leviathan.

Further reading II

Nash, J. F. (1950).
Non-cooperative games.
PhD thesis, Princeton University.

Nissen, V. (1997).
Einführung in evolutionäre Algorithmen: Optimierung nach dem Vorbild der Evolution.
Vieweg, Braunschweig/Wiesbaden, Germany.

Rechenberg, I. (1973).
Evolutionstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution.
Fromman-Holzboog Verlag, Stuttgart, Germany.