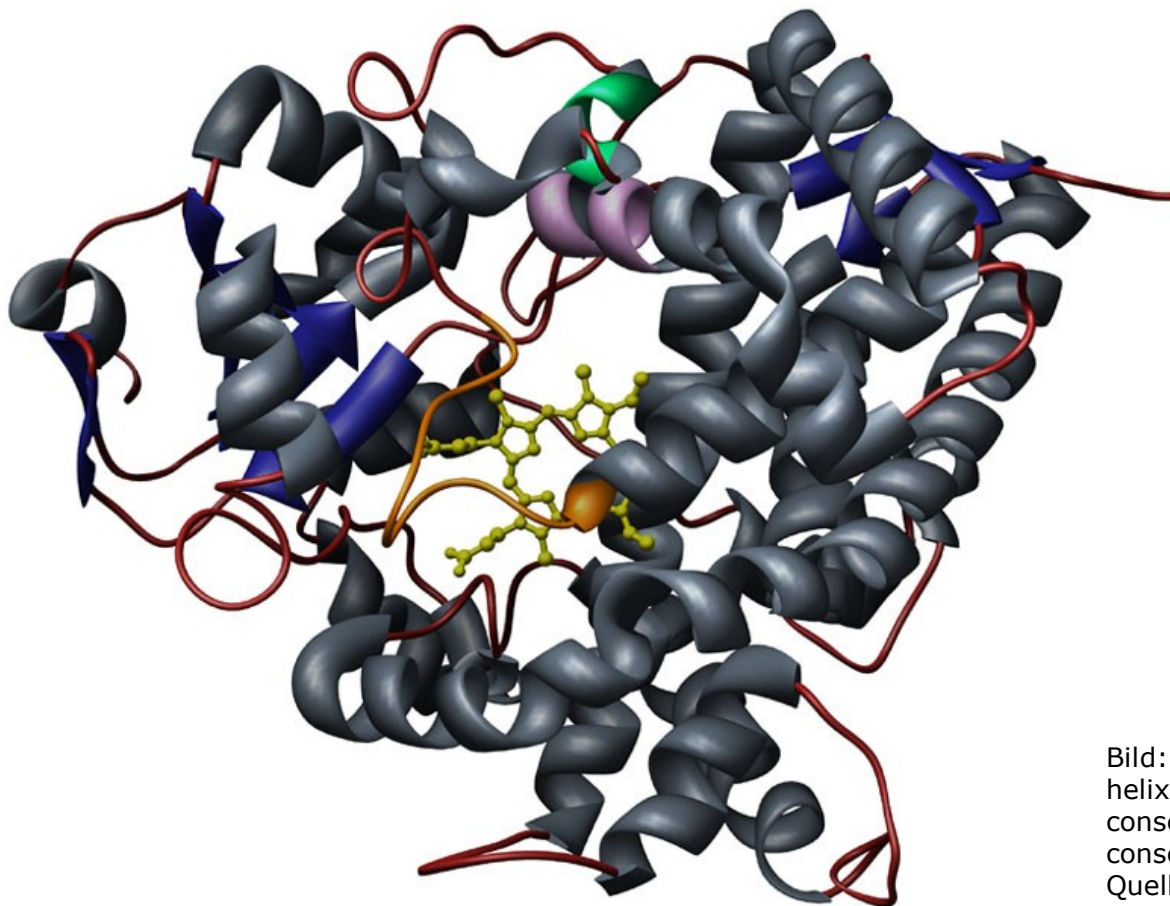


Prof. Dr. Alexander del Pino
Fachbereich Informatik
Sommersemester 2007

Genetische Algorithmen



7. Teil

Genetische
Operatoren

Bild: CYP3A4 [1TQN, Homo sapiens] ribbon detail. Color key:
helix structures, gray; strand structures, blue; PERF
consensus, green; K-helix consensus, purple; heme-binding
consensus, orange; heme ligand, yellow
Quelle: <http://p450.kvl.dk/gallery/CYP3A4.jpg>

Genetische Operatoren

Worum geht es ?

Ausgewählte Lösungskandidaten einer Population („Eltern“) werden durch genetische Operatoren verändert, um neue Lösungskandidaten („Nachkommen“) zu erhalten.

- Bei der **Mutation** entsteht aus genau einem Lösungskandidaten durch *zufällige Änderung* des Genotyps ein neuer Lösungskandidat. Die Mutation ist also ein unärer Operator.
- Bei dem **Crossover** entstehen aus zwei Lösungskandidaten durch *Austausch* deren Genotypen zwei neue Lösungskandidaten.

Genetische Operatoren hängen von der Kodierung der Lösungskandidaten ab.

Beispiel

Ein Crossover-Operator sieht anders aus, wenn der Lösungskandidat eine Integerzahl, eine Gleitkommazahl oder gar eine Permutation darstellt.

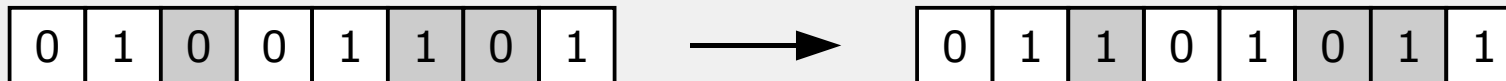
Genetische Operatoren

Mutation von Bitstrings

Bei einem Bitstring wird durch eine Mutation jedes Bit unabhängig von den anderen Bits mit einer Wahrscheinlichkeit p_m invertiert.

Beispiel

Bei einem 8-Bit-Vektor werden die Bits B1, B2 und B5 mutiert:



Genetische Operatoren

Mutation von Integerzahlen – Random Resetting

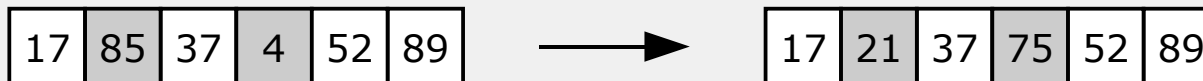
Bei einer Kodierung mit Integerzahlen wird häufig eine der beiden folgenden Mutationen verwendet. In beiden Fällen wird jede Integerzahl unabhängig von den anderen Zahlen mit einer Wahrscheinlichkeit p_m mutiert.

Bei dem *random resetting* wird eine Zahl durch eine Zufallszahl aus dem gültigen Wertebereich ersetzt.

Beispiel

Ein Chromosom kodiert sechs Winkelangaben von 0 bis 90 Grad.

Der dritte und fünfte Wert werden durch random resetting mit Zufallszahlen ersetzt.



Genetische Operatoren

Mutation von Integerzahlen – Creep Mutation

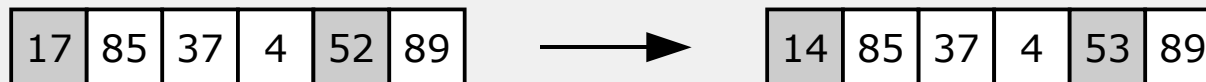
Bei der *creep mutation* wird auf die zu mutierende Zahl eine kleine Zahl addiert oder subtrahiert.

Im Gegensatz zu dem vorher betrachteten *random resetting* wird hier die zu mutierende Zahl mehr oder weniger stark verändert, die Mutation findet also gewissermassen schleichend (engl.: to creep = schleichen) statt.

Beispiel

Ein Chromosom kodiert sechs Winkelangaben von 0 bis 90 Grad.

Der zweite und der sechste Wert werden durch creep mutation um +1 bzw. um -3 verändert.



Genetische Operatoren

Mutation von Gleitkommazahlen

Beide eben vorgestellten Konzepte können auf Gleitkommazahlen erweitert werden:

- Analog zu dem *random resetting* wird eine Gleitkommazahl durch eine Zufallszahl ersetzt. Auch hier muss man aufpassen, dass die Zahlen oftmals nur in einem bestimmten Intervall $[\text{min}, \dots, \text{max}]$ gültig sind und die Zufallszahlen ebenfalls nur aus diesem Bereich stammen.
- Analog zu der *creep mutation* kann man bei Gleitkommazahlen ebenfalls eine kleine Zahl addieren oder subtrahieren um eine Mutation hervorzurufen. Diese kleine Zahl wird aus einer um Null zentrierten Normalverteilung mit einer gewünschten Standardabweichung gewählt.

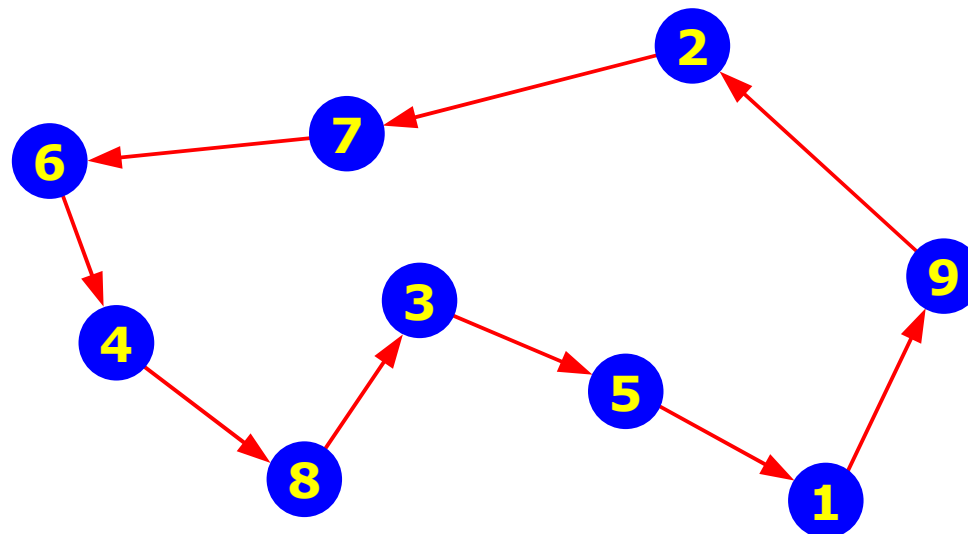
Genetische Operatoren

Mutation von Permutationen

Im Gegensatz zu den bisher betrachteten Mutationen von Zahlen ist es bei einer Permutation nicht mehr möglich, die Gene einzeln zu betrachten.

Beispiel

Bei dem Problem des Handlungsreisenden kodiert ein Lösungskandidat eine mögliche Rundreise durch mehrere Städte. Bei der Mutation muss ausgeschlossen werden, dass dadurch ein Lösungskandidat entsteht, bei dem eine Stadt mehrmals besucht wird.



Genetische Operatoren

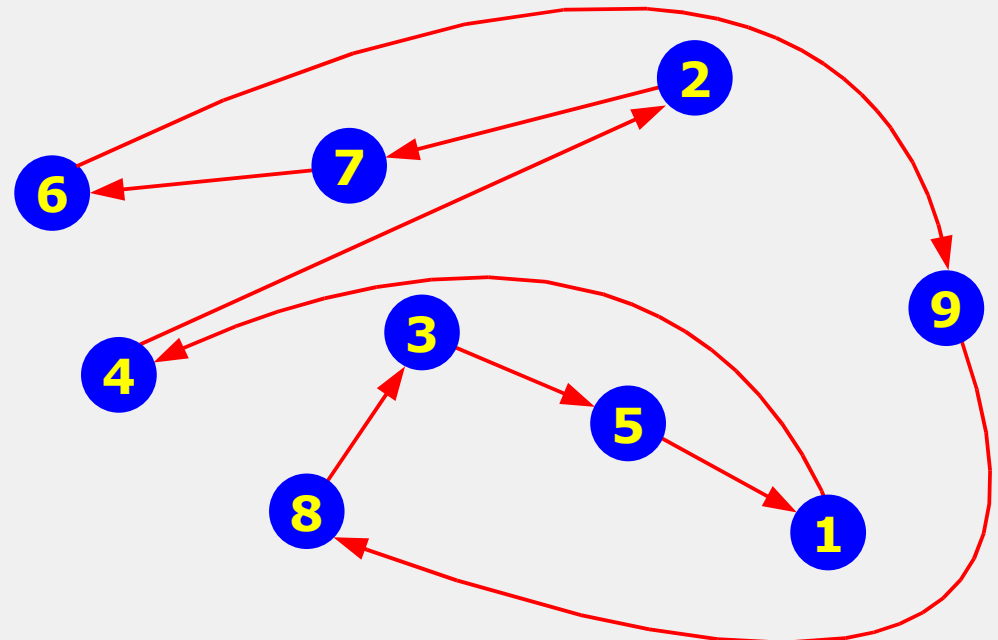
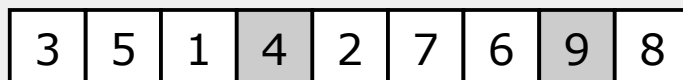
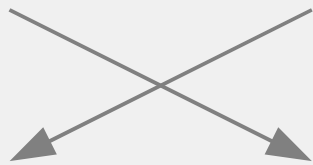
Mutation von Permutationen – Paarweiser Platztausch

Bei dem *paarweisen Platztausch* (*swap mutation*) werden zwei zufällig ausgesuchte Gene vertauscht.

Hierbei wird vorausgesetzt, dass die beiden Gene die gleichen Allelmengen besitzen.

Beispiel

Mutation einer pfadkodierten Rundreise durch paarweisen Platztausch.



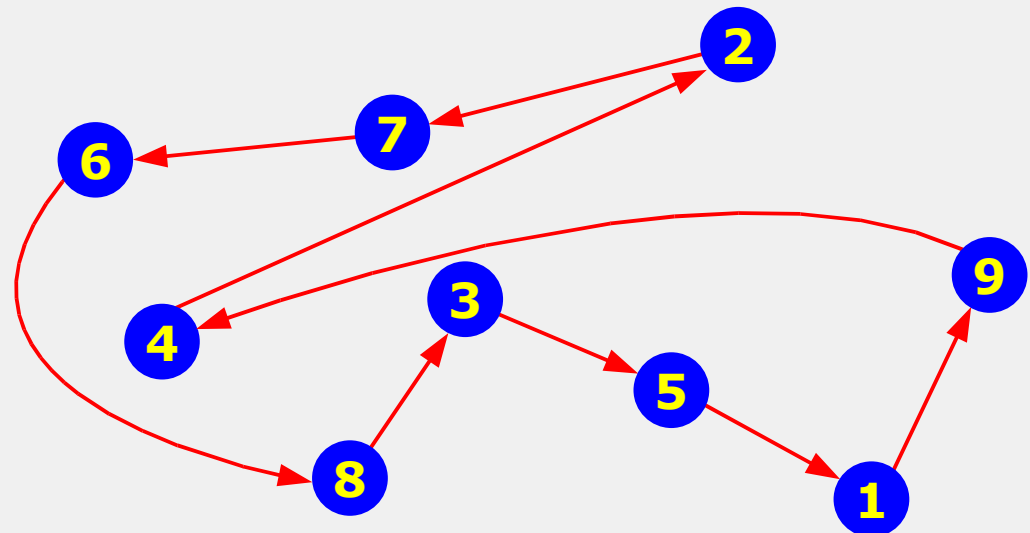
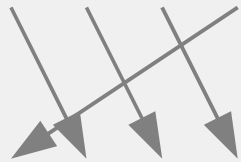
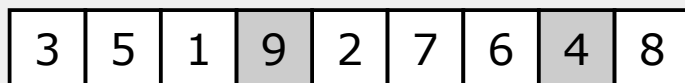
Genetische Operatoren

Mutation von Permutationen – Einfügemutation

Bei der *Einfügemutation* (*insert mutation*) wird ein Gen aus dem Chromosom entfernt und neben einem anderen Gen wieder eingefügt. Dabei verschieben sich die Gene auf einem Teilstück des Chromosoms.

Beispiel

Einfügemutation bei einer pfadkodierten Rundreise.



Genetische Operatoren

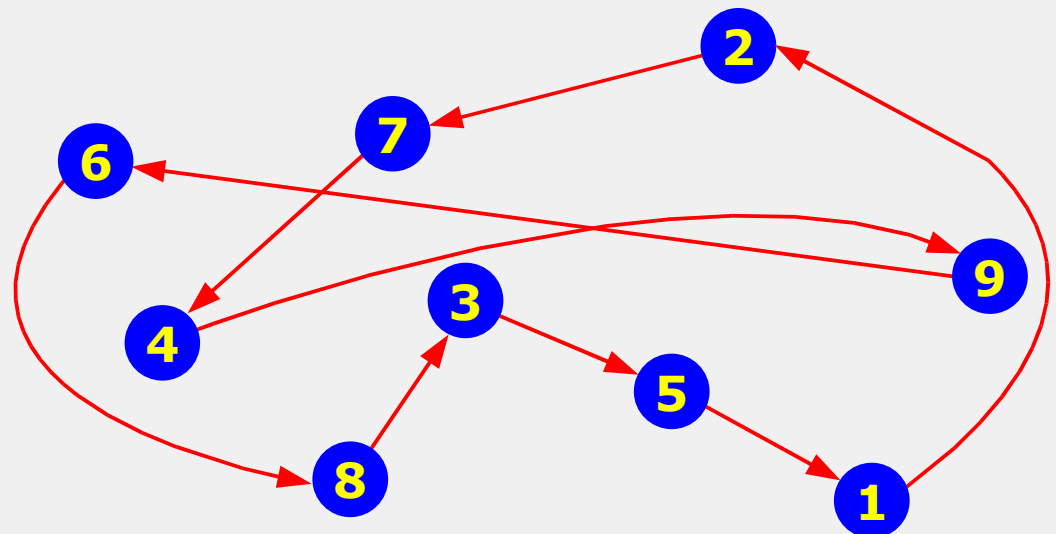
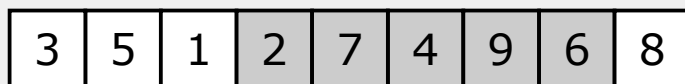
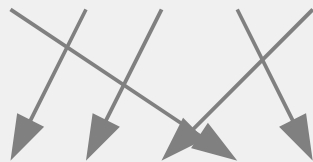
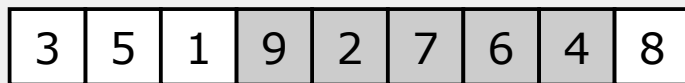
Mutation von Permutationen – Permutation

Ein Teilstück eines Chromosoms kann selbst wiederum durch eine **Permutation** umgestellt werden. Diese Art von Mutation ist auch unter dem Namen *scramble mutation* bekannt.

Dabei müssen alle betroffenen Gene des Teilstücks die gleiche Allelmenge besitzen.

Beispiel

Das grau markierte Teilstück einer pfadkodierten Rundreise wird durch eine Permutation mutiert.



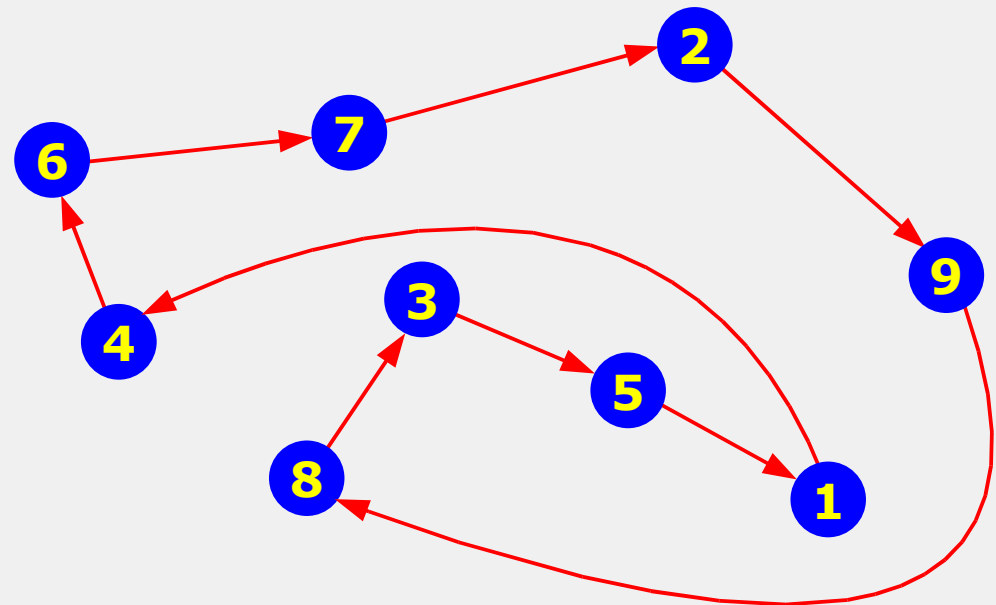
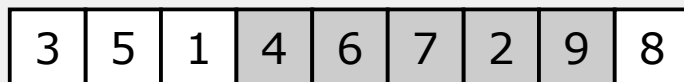
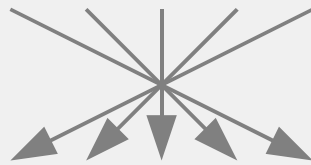
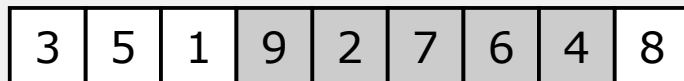
Genetische Operatoren

Mutation von Permutationen – Inversion

Bei der *Inversion* (*inversion mutation*) werden zwei Gene eines Chromosoms zufällig ausgewählt und alle Gene des so definierten Teilstücks herumgedreht.

Beispiel

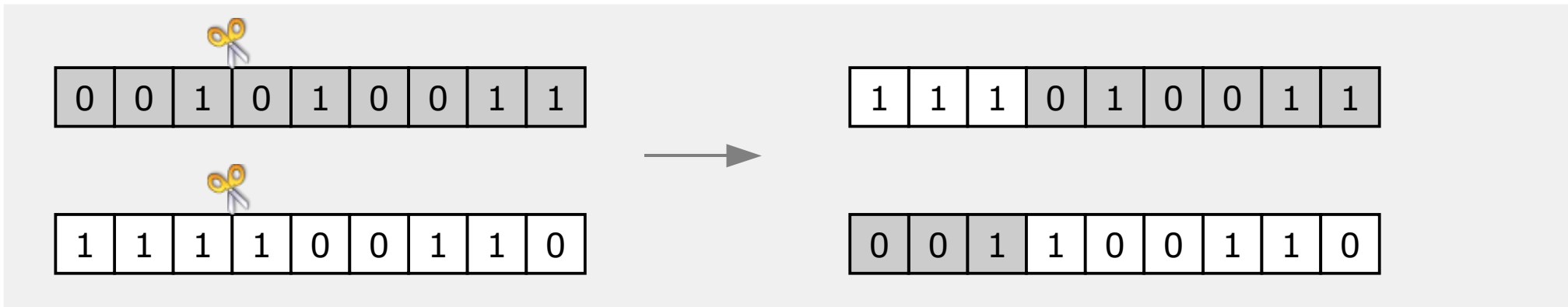
Inversion des grau markierten Teilstücks einer pfadkodierte Rundreise.



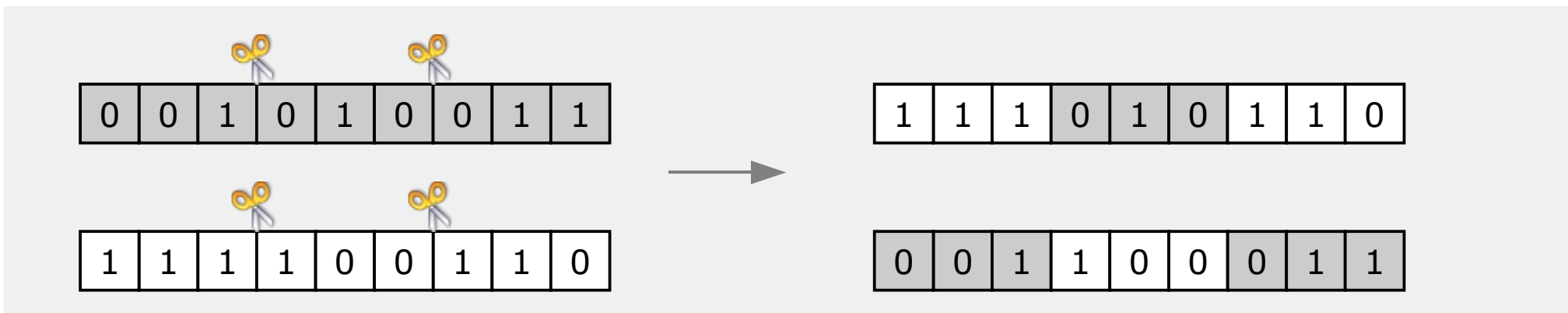
Genetische Operatoren

Crossover bei binären Codierungen

Den *1-Punkt-Crossover*-Operator (*1-point crossover*) haben wir bereits kennengelernt. Beispiel:



Der *N-Punkt-Crossover* Operator (*n-point crossover*) ist eine Verallgemeinerung hiervon. Beispiel mit N=2:



Genetische Operatoren

Uniformer Crossover bei binären Kodierungen

Bei dem *uniformen Crossover* (*uniform crossover*) wird für jedes Gen unabhängig von den anderen Genen auf dem Chromosom per gleichverteilte Zufallsvariable entschieden, ob das Gen von dem ersten oder dem zweiten Elternteil stammen soll.

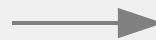
Beispiel

Bei einem Münzwurf hat jedes Gen die gleiche Wahrscheinlichkeit $p_c = 0,5$ von einem bestimmten Elternteil zu stammen, es wird also kein Elternteil bevorzugt.

0	0	1	0	1	0	0	1	1
---	---	---	---	---	---	---	---	---

K	Z	K	K	Z	K	Z	Z	K
---	---	---	---	---	---	---	---	---

1	1	1	1	0	0	1	1	0
---	---	---	---	---	---	---	---	---



0	1	1	0	0	0	1	1	1
---	---	---	---	---	---	---	---	---

1	0	1	1	1	0	0	1	0
---	---	---	---	---	---	---	---	---

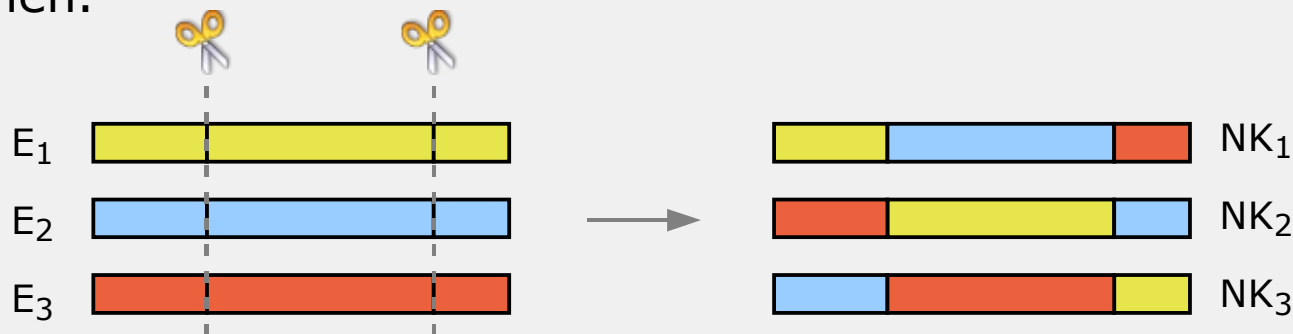
Genetische Operatoren

Diagonal-Crossover / Multi-Parent Recombination

A. Eiben schlug mit dem *Diagonal-Crossover* eine Verallgemeinerung des *1-Punkt-Crossover* Operators mit mehreren Elternteilen (*multi-parent recombination*) vor. Bei N Elternteilen werden die Chromosomen dabei an $N-1$ Stellen aufgetrennt.

Beispiel

Diagonal-Crossover mit drei Elternteilen, aus denen drei Nachkommen entstehen.



Dieser Operator zeigt, wie bei genetischen Algorithmen auch Mechanismen eingesetzt werden können, die kein Äquivalent in der biologischen Evolution besitzen.

Genetische Operatoren

Positionsabhängige Verzerrungen bei Crossover-Operatoren

Bei dem *N-Punkt-Crossover* Operator besteht die Tendenz, dass auf dem Chromosom benachbarte Gene nach der Anwendung des Operators weiterhin zusammen bleiben.

Ebenso besteht die Tendenz, dass bei ungeradem N eine Gengruppe die an beiden Enden des Chromosoms positioniert ist, aufgebrochen wird.

Solche Effekte nennt man *positionsabhängige Verzerrungen* (*positional bias*).

❓ Warum ausgerechnet bei ungeradem N ?

Genetische Operatoren

Verteilungsabhängige Verzerrungen bei Crossover-Operatoren

Bei dem uniformen Crossover gibt es diese positionsabhängigen Verzerrungen nicht, aber es besteht die Tendenz, dass die Nachkommen meistens etwa 50% der Gene der jeweiligen Elternteile erhalten.

Wenn das Elternteil eines Gens mit einer Wahrscheinlichkeit p_c ermittelt wird, ist die Anzahl k der insgesamt erhaltenen Gene von einem Elternteil wie folgt binomial verteilt:

$$P(\text{Anzahl} = k) = \binom{n}{k} * p_c^k * (1 - p_c)^{n-k}$$

Es ist also eher unwahrscheinlich, dass von einem Elternteil sehr viele und von dem anderen Elternteil sehr wenig Gene vererbt werden. Diesen Effekt nennt man *verteilungsabhängige Verzerrung (distributional bias)*.

Genetische Operatoren

Crossover bei Gleitkommazahlen

Die eben vorgestellten Crossover-Operatoren lassen sich auf Gleitkommazahlen verallgemeinern. Anstatt ein Bit oder eine Integerzahl wird dann eine Gleitkommazahl betrachtet. Diese Vorgehensweise nennt man *discrete recombination*.

Bei den *arithmetischen Crossover-Operatoren* (*arithmetic recombination*) werden für ein bestimmtes Gen die Allele A_1 und A_2 der beiden Elternteile mit einem Faktor f gewichtet um die neuen Allele NK_1 und NK_2 für die Nachkommen zu erhalten.

$$NK_1 = f * A_1 + (1 - f) * A_2$$

$$NK_2 = f * A_2 + (1 - f) * A_1$$

- ❓ Welchen Wert sollte f haben um die Allele der Elternteile gleich zu gewichten ? Was ist die Konsequenz daraus ?

Genetische Operatoren

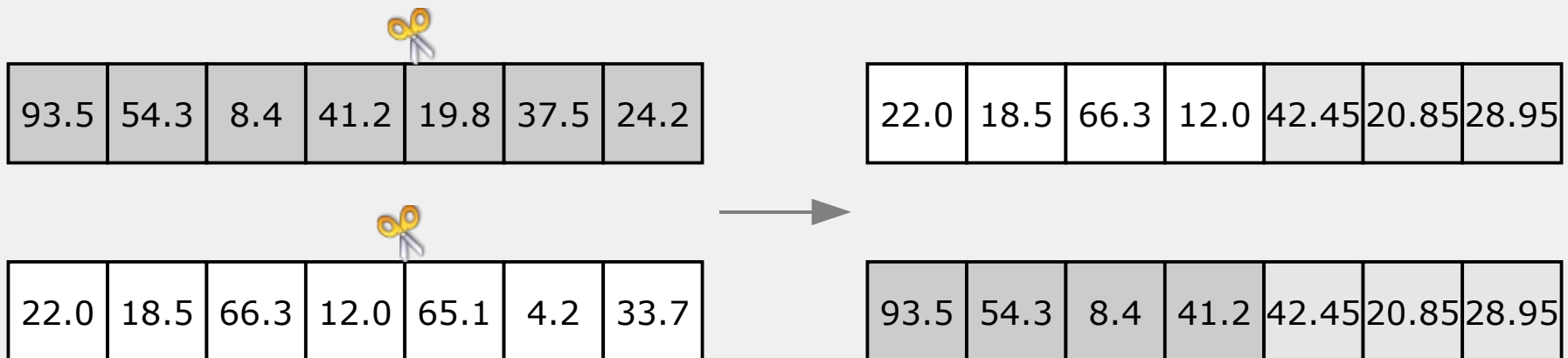
Crossover bei Gleitkommazahlen

Bei der *einfachen Rekombination* (*simple recombination*) von n Gleitkommazahlen wird zuerst per Zufallszahl ein Index $k < n$ ermittelt.

Die ersten k Gene des ersten Nachkommen stammen von dem zweiten Elternteil. Die verbleibenden $n-k$ Gene des ersten Nachkommen werden wie eben beschrieben durch arithmetische Gewichtung bestimmt.

Beispiel:

$n=7$, nehmen wir an dass $k=4$ und $f=0.5$ ist.



$$0.5 * 19.8 + (1 - 0.5) * 65.1 = 42.45$$

Genetische Operatoren

Crossover bei Gleitkommazahlen

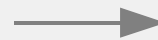
Bei der *single arithmetic recombination* wird lediglich ein einziges per Zufall ermitteltes Gen auf dem gesamten Chromosom durch arithmetische Gewichtung bestimmt, alle anderen Gene bleiben erhalten:

Beispiel

$n=7$, angenommen $k=3$ und $f=0.5$

93.5	54.3	8.4	41.2	19.8	37.5	24.2
------	------	-----	------	------	------	------

93.5	54.3	37.35	41.2	19.8	37.5	24.2
------	------	-------	------	------	------	------



22.0	18.5	66.3	12.0	65.1	4.2	33.7
------	------	------	------	------	-----	------

22.0	18.5	37.35	12.0	65.1	4.2	33.7
------	------	-------	------	------	-----	------

$$0.5 * 8.4 + (1-0.5) * 66.3 = 37.35$$

Genetische Operatoren

Crossover bei Gleitkommazahlen

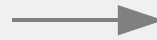
Bei der *whole arithmetic recombination* werden alle Gene der beiden Nachkommen durch arithmetische Gewichtung bestimmt.

Beispiel

$n=7$ und $f=0.4$

93.5	54.3	8.4	41.2	19.8	37.5	24.2
------	------	-----	------	------	------	------

22.0	18.5	66.3	12.0	65.1	4.2	33.7
------	------	------	------	------	-----	------



50.6	32.82	43.14	23.68	46.98	17.52	29.9
------	-------	-------	-------	-------	-------	------

64.9	39.98	31.56	29.52	37.92	24.18	28
------	-------	-------	-------	-------	-------	----

Beachten Sie, dass in diesem Beispiel f nicht mehr 0.5 ist, und somit bei dem Crossover zwei verschiedene Allele für jedes Gen entstehen:

$$0.4 * 93.5 + (1-0.4) * 22.0 = 50.6$$

$$0.4 * 22.0 + (1-0.4) * 93.5 = 64.9$$

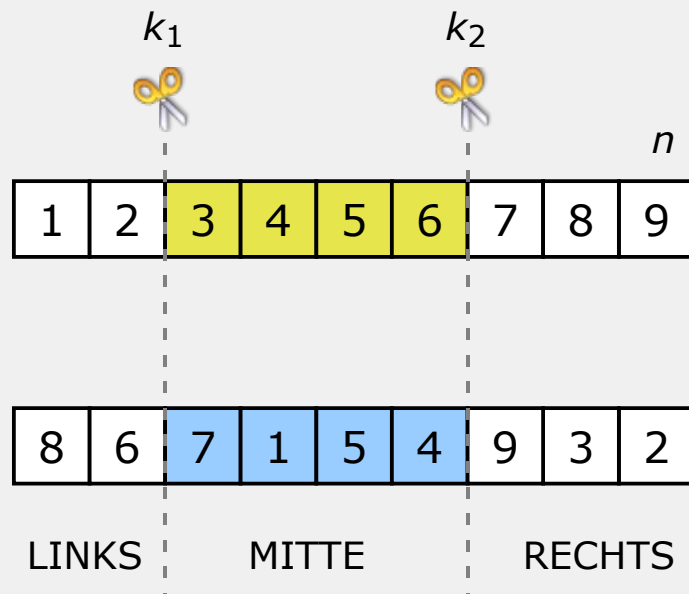
Genetische Operatoren

Crossover bei Permutationen – Partially Mapped Crossover

Partially mapped crossover (PMX) wurde 1985 von *Goldberg* und *Lingle* vorgeschlagen.

- Zuerst werden zwei Positionen $0 < k_1 < k_2 < n$ per Zufallszahl ermittelt.
- Anschließend werden die beiden Elternteile an diesen Positionen in drei Segmente LINKS, MITTE und RECHTS aufgetrennt.

Beispiel



Genetische Operatoren

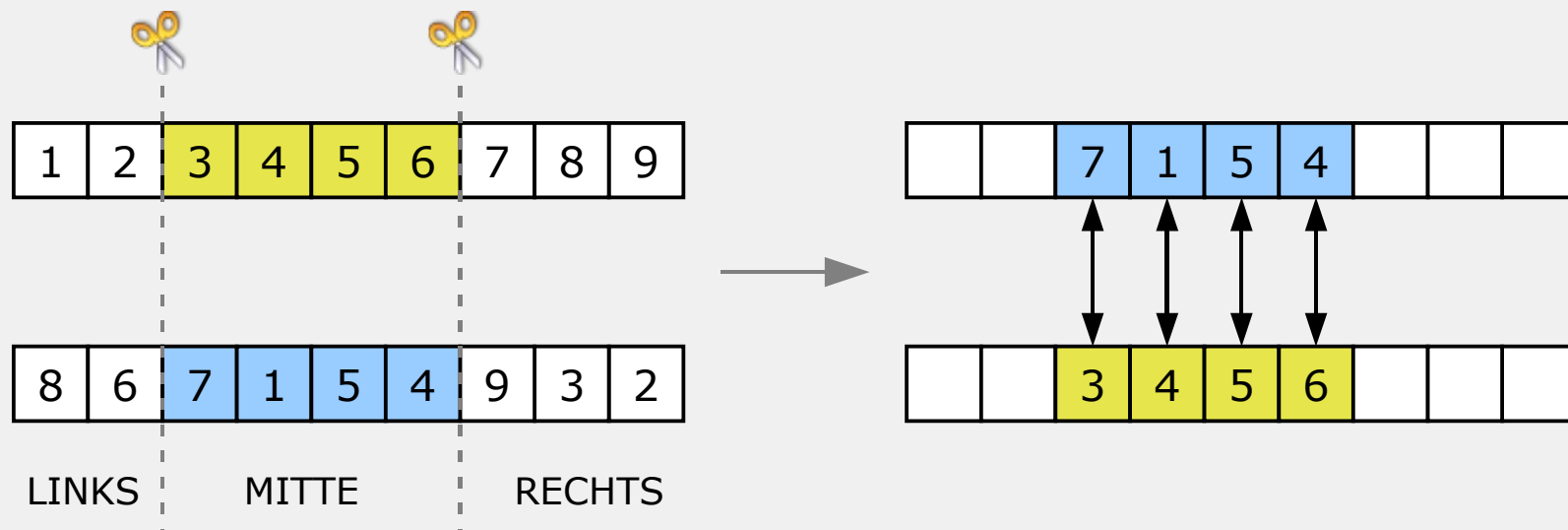
Crossover bei Permutationen – Partially Mapped Crossover

Als nächstes erhalten die Nachkommen jeweils ein MITTE Segment.

Damit werden Abbildungen für die Gene im MITTE-Segment definiert.

Beispiel

Die Abbildungen lauten hier $7 \leftrightarrow 3$, $1 \leftrightarrow 4$, $5 \leftrightarrow 5$ und $4 \leftrightarrow 6$.



Genetische Operatoren

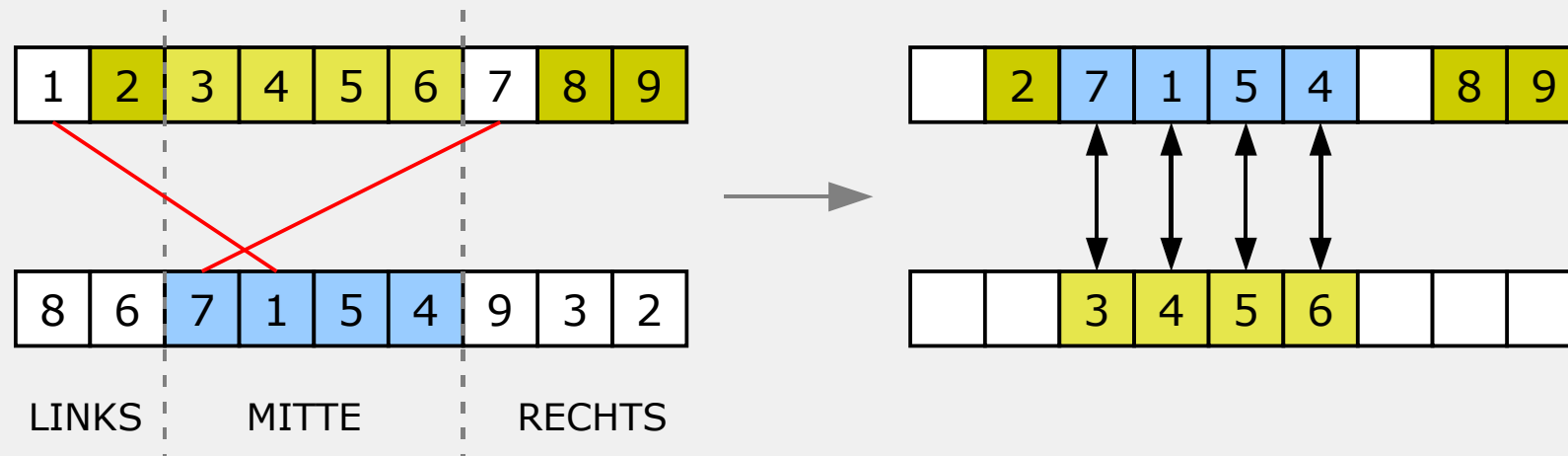
Crossover bei Permutationen – Partially Mapped Crossover

Im nächsten Schritt werden diejenigen Gene aus den LINKS- und RECHTS-Segmenten, die nicht im gegenüber liegenden MITTE-Segment enthalten sind, an die gleiche Position bei den Nachkommen kopiert.

Beispiel

Gen 1 und Gen 7 sind im gegenüber liegenden MITTE-Segment enthalten und können nicht kopiert werden (durch die roten Linien angedeutet).

Gen 2, Gen 8 und Gen 9 können problemlos kopiert werden.



Genetische Operatoren

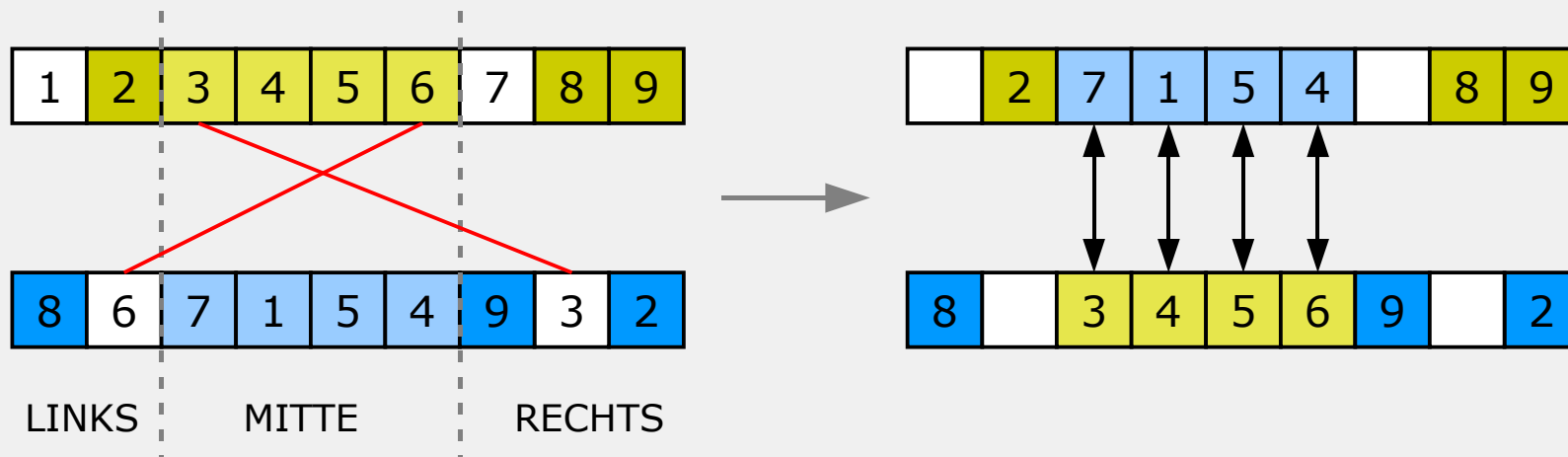
Crossover bei Permutationen – Partially Mapped Crossover

Beispiel

Nun die gleiche Prozedur für den anderen Nachkommen.

Gen 6 und Gen 3 sind im gegenüber liegenden MITTE-Segment enthalten und können nicht kopiert werden (durch die roten Linien angedeutet).

Gen 8, Gen 9 und Gen 2 können problemlos kopiert werden.



Genetische Operatoren

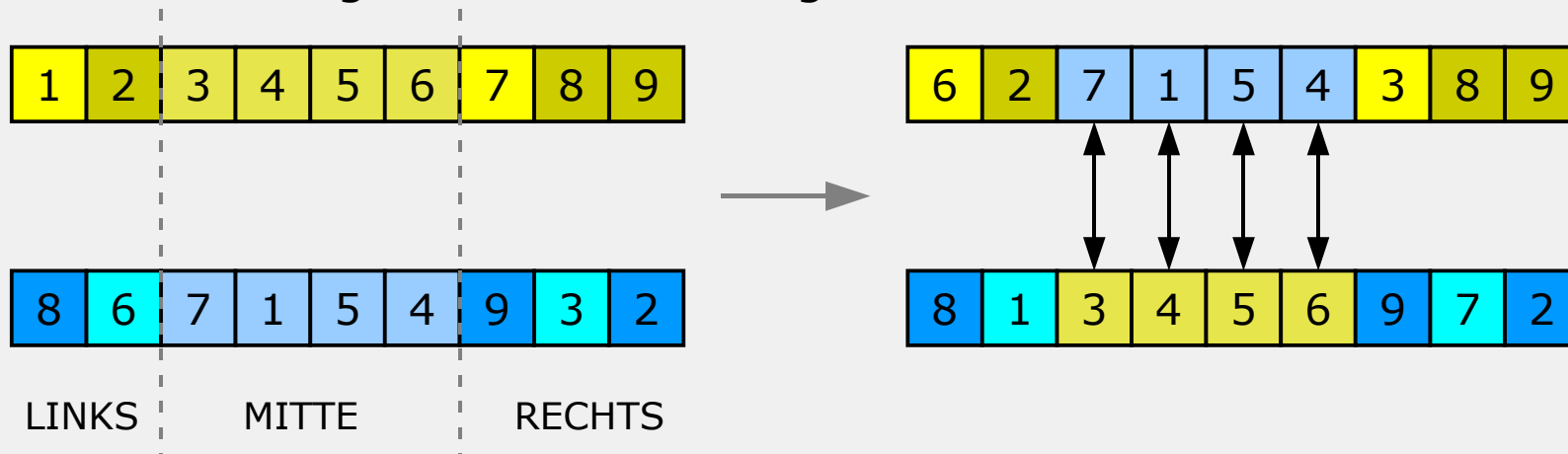
Crossover bei Permutationen – Partially Mapped Crossover

Im letzten Schritt werden die Gene für die noch freien Stellen im Chromosom gegebenenfalls durch mehrfachen Lookup der vorhin festgelegten Abbildung ermittelt.

Beispiel

Gen 1 ist ja im gegenüber liegenden MITTE-Segment enthalten. Durch mehrfachen Lookup $1 \leftrightarrow 4 \leftrightarrow 6$ wird Gen 6 ermittelt.

Das Gesamtergebnis sieht wie folgt aus:



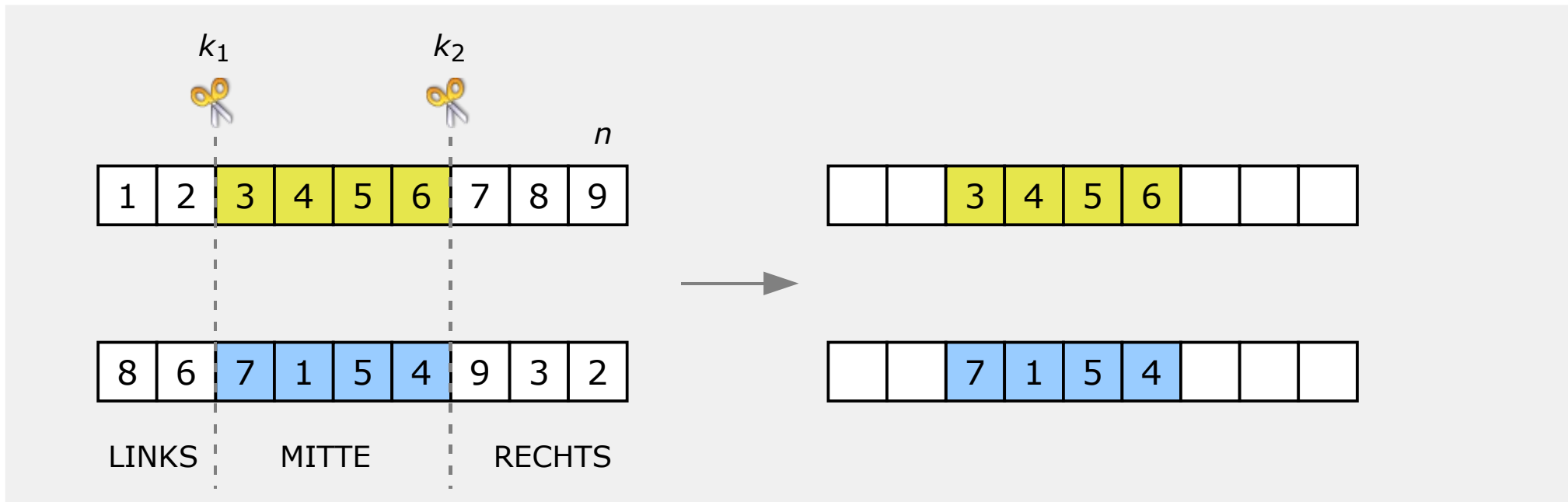
Genetische Operatoren

Crossover bei Permutationen – Order Crossover

Order crossover (OX) wurde von *Davis* als ein Operator vorgeschlagen, der die Anordnung der Elemente in einer Permutation weitgehend beibehält.

Order crossover beginnt ähnlich wie *partially mapped crossover* mit dem Kopieren eines MITTE-Segments in die Nachkommen.

Beispiel



Genetische Operatoren

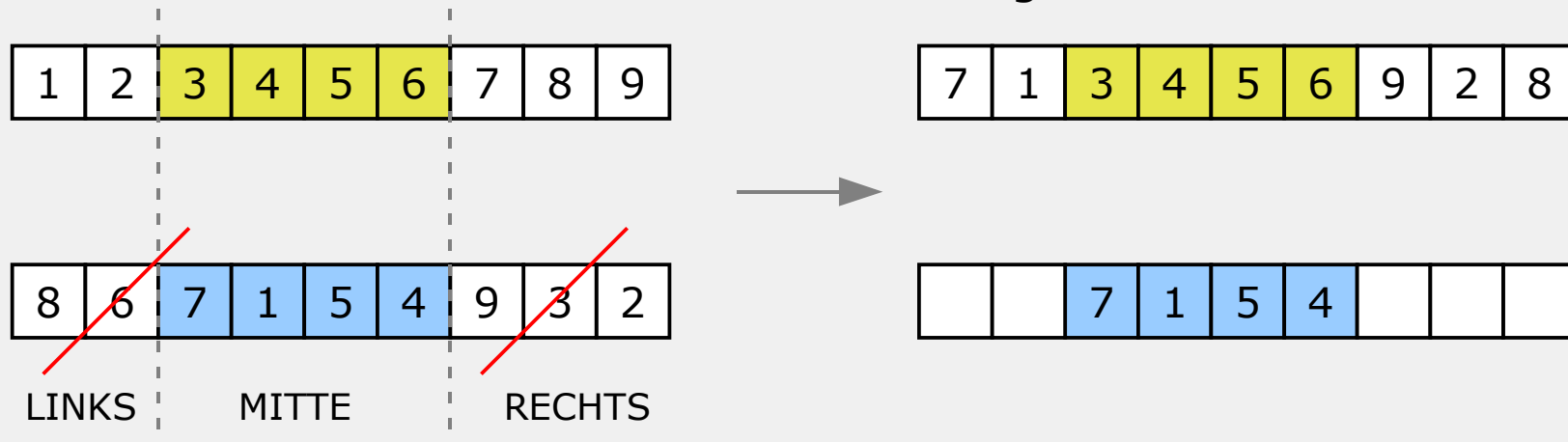
Crossover bei Permutationen – Order Crossover

Im nächsten Schritt werden die freien Loci mit den Genen aus dem jeweils anderen Elternteil aufgefüllt.

- Das Auffüllen beginnt mit dem RECHTS-Segment.
- Gene, die bereits im MITTE-Segment enthalten sind werden ignoriert.
- Wenn man mit dem Auffüllen am rechten Ende des RECHTS-Segments angekommen ist, macht man mit beim LINKS-Segment weiter.

Beispiel

Das Auffüllen des oberen Nachkommens beginnt mit Gen 9.



Genetische Operatoren

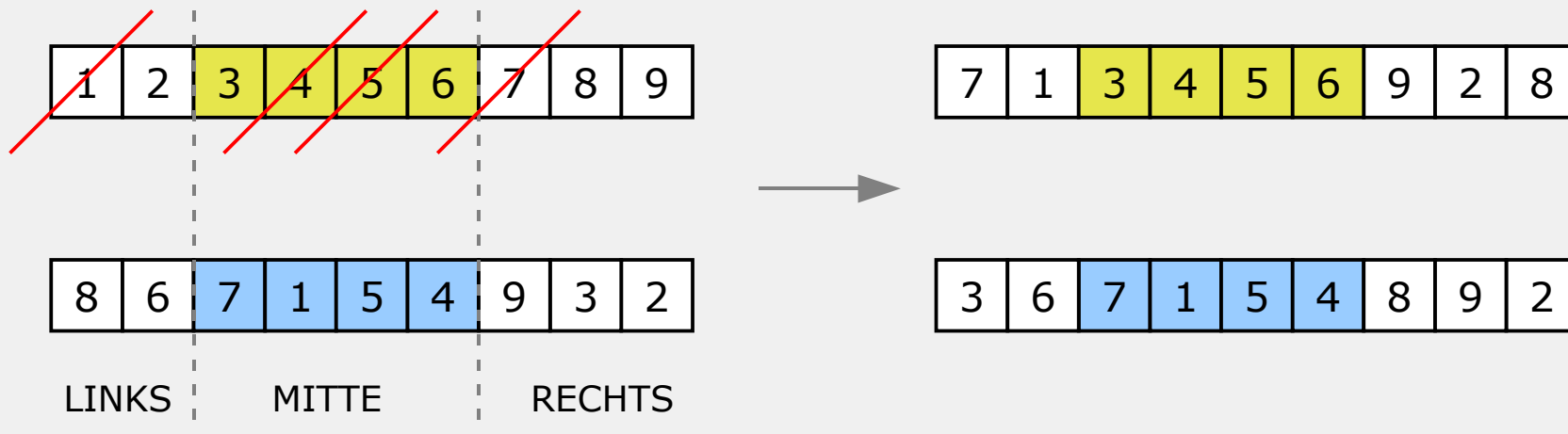
Crossover bei Permutationen – Order Crossover

Beispiel

Das Auffüllen des unteren Nachkommens beginnt mit Gen 7, welches ignoriert wird, da es bereits im MITTE-Segment vorhanden ist.

Danach geht es weiter mit den Genen 8 und 9.

Das Ergebnis sieht wie folgt aus:



Genetische Operatoren

Crossover bei Permutationen – Cycle Crossover

Cycle crossover wurde von *I. Oliver* vorgeschlagen und versucht, so weit wie möglich die absoluten Positionen der Gene beizubehalten.

Bei dem *cycle crossover* wird das Chromosom in Zyklus aufgetrennt, die dann abwechselnd beibehalten oder ausgetauscht werden.

Zuerst bestimmt man also wie folgt alle Zyklen:

1. Ein Zyklus beginnt mit der ersten freien Allele A_i des ersten Elternteils.
2. Betrachte die Allele B_i des zweiten Elternteils an der gleichen Position.
3. Suche die Position j wo sich B_i im ersten Elternteil befindet.
4. Füge die Allele A_j dem Zyklus hinzu und setze $i=j$.
5. Wiederhole Schritte 2-4 bis der Zyklus geschlossen ist.

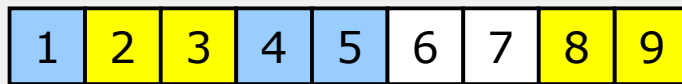
Genetische Operatoren

Crossover bei Permutationen – Cycle Crossover

Beispiel

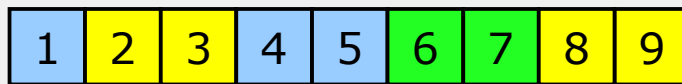


Der erste Zyklus lautet $1 \rightarrow 4 \rightarrow 5 \rightarrow 1 \dots$



Der zweite Zyklus lautet $2 \rightarrow 8 \rightarrow 9 \rightarrow 3 \rightarrow 2$

...



Der dritte Zyklus lautet $6 \rightarrow 7 \rightarrow 6 \dots$



Genetische Operatoren

Crossover bei Permutationen – Cycle Crossover

Wenn alle Zyklen bekannt sind, wird das Chromosom an den Zyklengrenzen aufgetrennt und die Segmente jedes zweiten Zyklus werden vertauscht.

Beispiel

In unserem Beispiel mit drei Zyklen bleibt Zyklus 1 (blau) erhalten, Zyklus 2 (gelb) wird ausgetauscht und Zyklus 3 (grün) bleibt wieder erhalten.

Somit ergibt sich folgendes Ergebnis:

