

# Übung 6

Ruman Gerst, 136994

May 16, 2014

## Aufgabe 1

### Umsetzung

Zusätzlich zu der gestellten Aufgabe wird verglichen, wie sich die Kodierungs- und Dekodierungsfunktionen im Vergleich zwischen Standard-Binärcode und Gray-Code verhalten. Als Funktion zur Bewertung der Abhängigkeit zwischen Phänotyp und Genotyp wird der Abstand des aktuellen Phänotyps von dem der letzten Generation durch die die Anzahl der veränderten Bits gegenüber der Vorgängergeneration geteilt, also  $\frac{\Delta phenotype(t,t-1)}{\Delta genotype(t,t-1)}$  mit  $\Delta phenotype(p_1, p) =$

$$|p - p_2| \text{ und } \Delta genotype(s_1, s_2) = \sum_i f(s_1[i], s_2[i]) \text{ mit } f(x, y) = \begin{cases} 1 & x \neq y \\ 0 & \text{sonst} \end{cases}.$$

Da Gray-Code mit Standard-Binärcode verglichen werden soll, werden bei der Mutation der beiden Bitstrings die  $u$  zuerst ausgerechnet und beide Mutationen mit den selben  $u$  ausgeführt. Dies stellt sicher, dass bei beiden Strings die selben Bits mutiert werden. Es wurde  $P = 0$  gesetzt, der Algorithmus führt die Mutation für 100 Generationen aus.

### Ergebnisse

Bereits bei einer sehr kleinen Mutationswahrscheinlichkeit  $p_m = 0.01$  wird deutlich, dass selbst kleine Mutationen im Genom eine sehr große Änderung im Phänotyp zur Folge haben kann (siehe Figure 1). So kann die bewertete Änderung bis zu 8000 oder sogar 16000 betragen, was wahrscheinlich durch ein mutiertes Bit in den ersten Stellen des Bitstrings ausgelöst wird. Der Average ist bei  $p_m = 0.01$  aber noch relativ klein.

Wie man an den Figures 1, 2 und 3 erkennt, steigt die average bewertete Änderung, je höher die Mutationsrate  $p_m$  angesetzt wird. Dies ist sinnvoll, da die

Wahrscheinlichkeit steigt, dass Bits zu einer 1 geflippt werden und damit die Zahl höher wird.

Beim Vergleich zwischen Binärcodierung und Gray-Code wird deutlich, dass sich beide Codierungen bei kleinen Mutationsraten sehr ähnlich verhalten. Der Gray-Code erzeugt sogar höhere Abweichungen (siehe Figure 1), wenn die Mutationsrate klein ist. Erst bei höheren Mutationsraten (siehe Figures 3, 4) hat der Gray-Code bessere Ergebnisse wie der Binärcode. So sind die Abweichungen des Gray-Codes bei  $p_m = 0.95$  schon deutlich kleiner als die des Binärcodes. Als Extrembeispiel wird  $p_m = 1$  gesetzt, was einem vollständigen Umdrehen der Bitstrings entspricht. Hier hat der Gray-Code eine deutlich geringere bewertete Abweichung zu dem Phänotyp der vorherigen Generation. Da immer gleich viele Bits mutiert werden, lässt sich der jeweilige Phänotyp errechnen.

Der Gray-Code ist also insgesamt weniger Störanfällig gegenüber Mutationen, besonders wenn eine hohe Mutationsrate vorliegt.

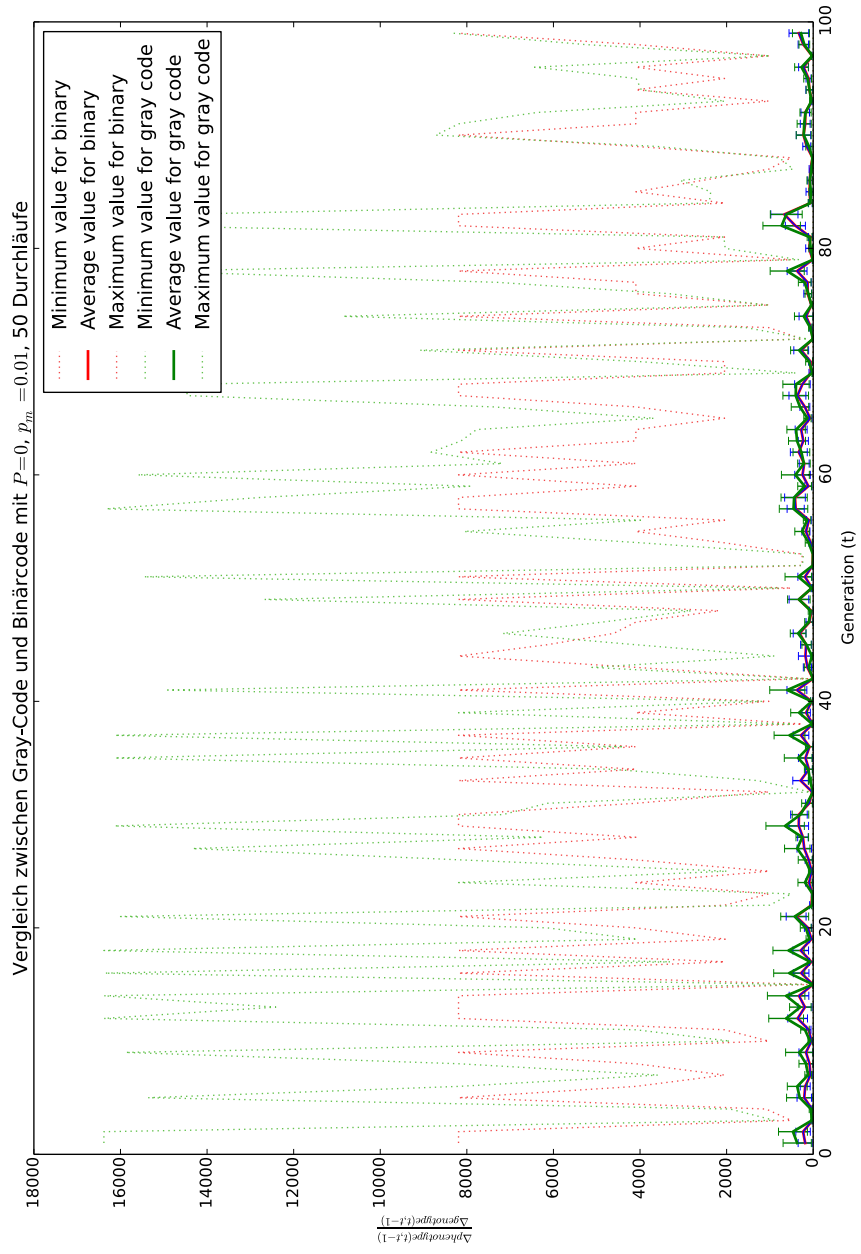


Figure 1: Test mit  $p_m = 0.01$

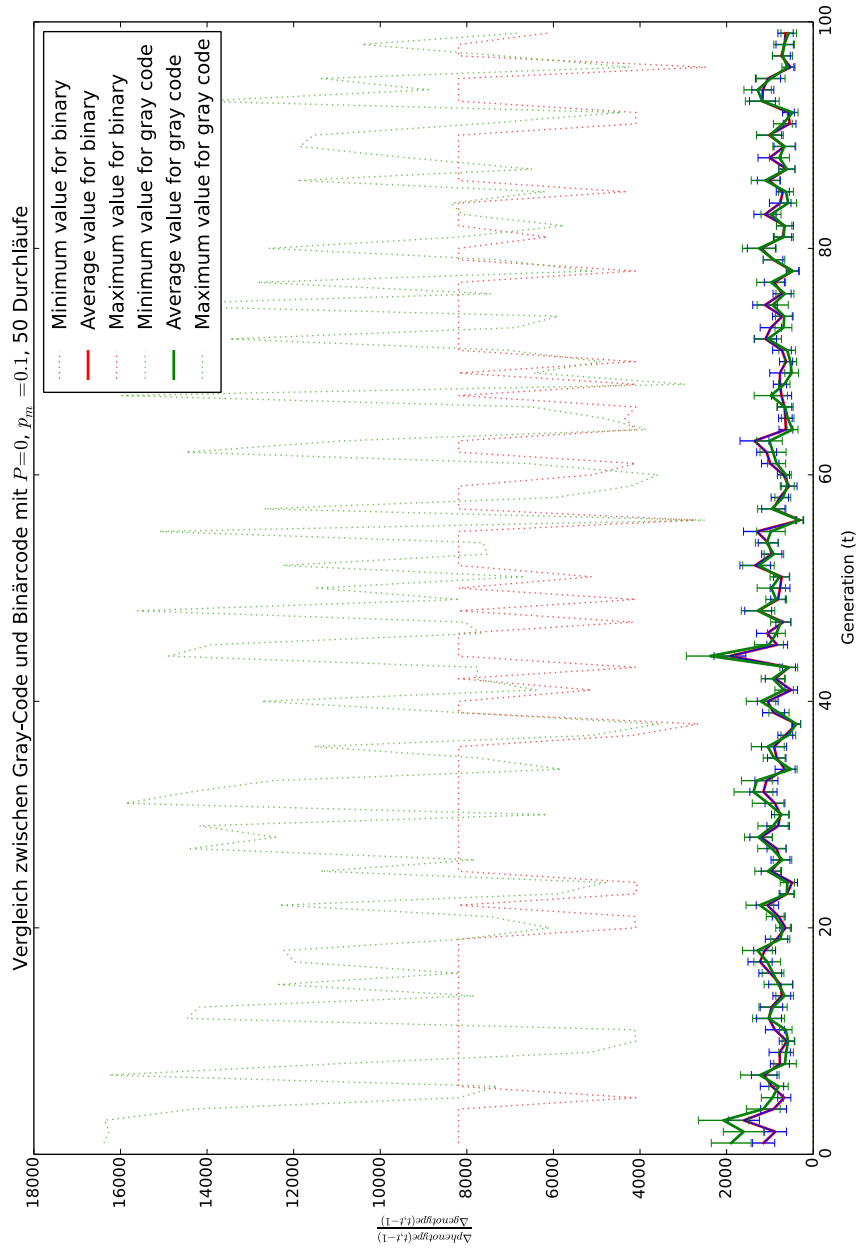


Figure 2: Test mit  $p_m = 0.1$

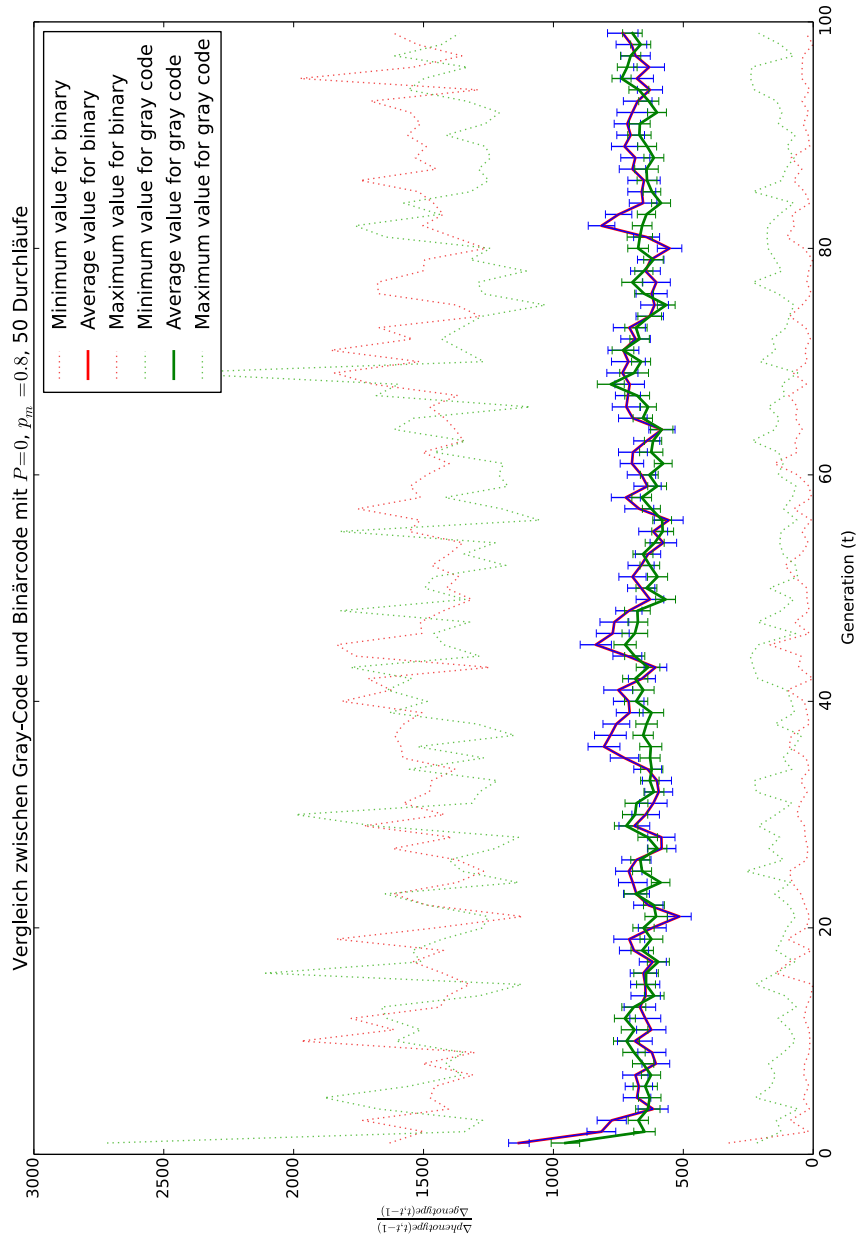


Figure 3: Test mit  $p_m = 0.8$

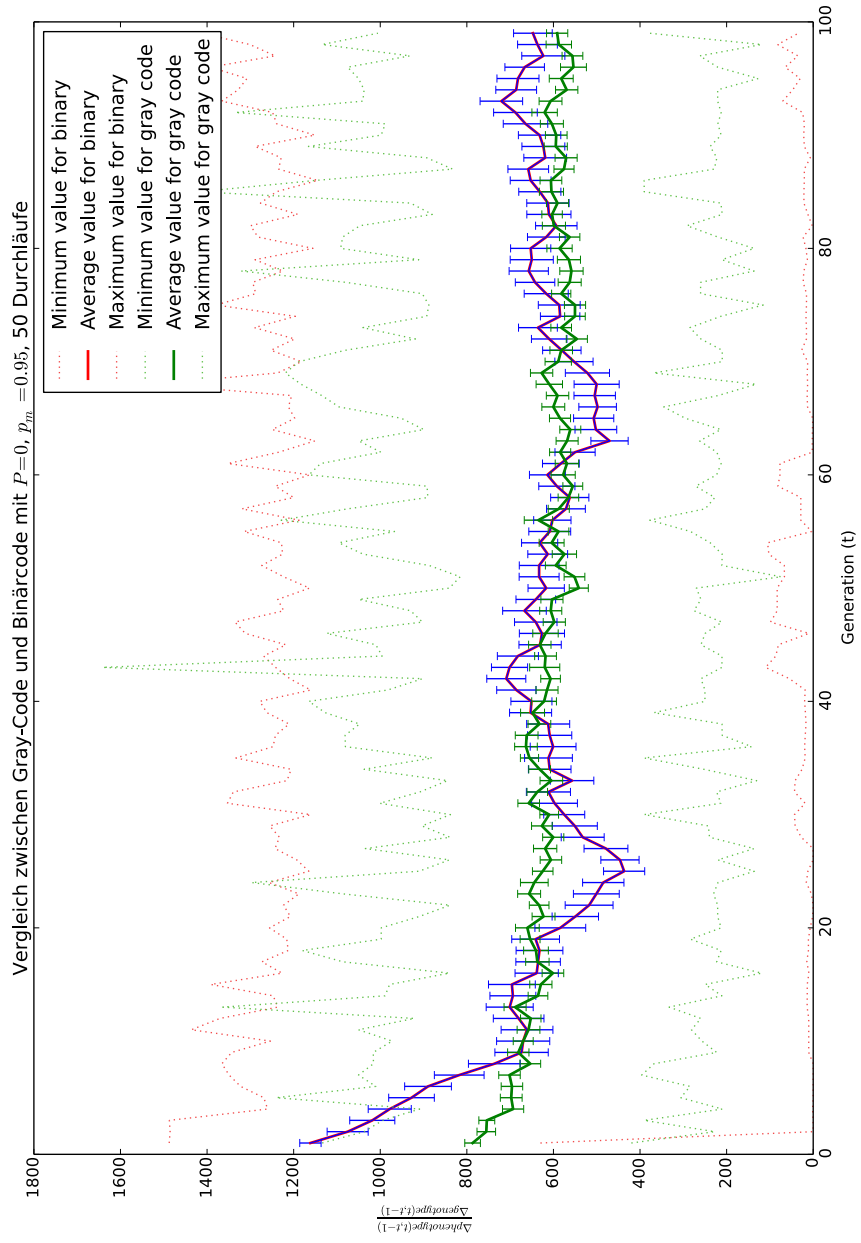


Figure 4: Test mit  $p_m = 0.95$

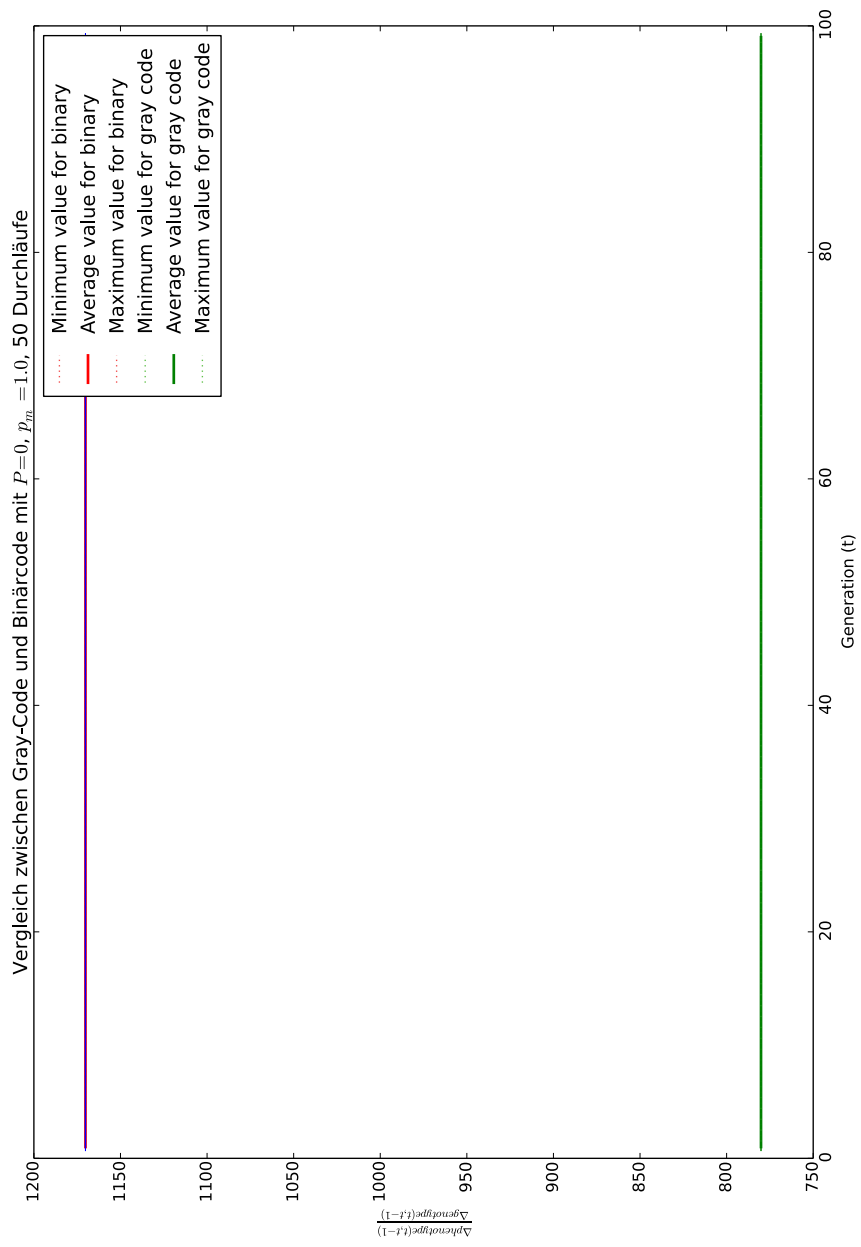


Figure 5: Test mit  $p_m = 1$