



Akademia Górniczo-Hutnicza
im. Stanisława Staszica w Krakowie



Wydział Fizyki i Informatyki Stosowanej

Symulacje Monte-Carlo procesów starzenia biologicznego i ewolucji populacji

Rozprawa doktorska

Mikołaj Sitarz

Promotor: prof. dr hab. Andrzej Maksymowicz

Praca przedstawiona

**Radzie Wydziału Fizyki i Informatyki Stosowanej
Akademii Górniczo-Hutniczej
w Krakowie**

Kraków, marzec 2006

Pragnę złożyć serdeczne podziękowania dla profesora Andrzeja Maksymowicza za niezliczone dyskusje, wiele cennych podpowiedzi, oraz całą okazaną mi pomoc, dzięki której praca ta mogła powstać.

Spis treści

1	Wstęp	6
1.1	Cele pracy	6
1.2	Możliwe przyczyny biologicznego starzenia	7
1.3	Symulacje Monte-Carlo	8
1.4	Wybrane modele rozwoju populacji	9
1.4.1	Model logistyczny	9
1.4.2	Model Penny	13
1.4.3	Model Penny z uwzględnieniem płci osobników	19
2	Przegląd badań nad procesami ewolucji populacji	21
2.1	Linie genetyczne	21
2.2	Zasadnicze wyniki w aseksualnym modelu Penny	22
2.2.1	Prawo Gompertza	22
2.2.2	Zestawienie wyników symulacji z danymi demograficznymi	24
3	Symulacje komputerowe	28
3.1	Podstawowy algorytm symulacji komputerowej	28
3.1.1	Opis implementacji algorytmu	28
3.1.2	Ocena warunków symulacji	29
3.1.3	Oszacowanie błędu wyników	33
3.2	Analiza słuszności prawa Gompertza dla różnych mechanizmów śmierci	35
3.3	Dynamika redukcji rodzin w modelu seksualnym i aseksualnym .	35
3.4	Rola selektywnego doboru partnera w modelu seksualnym	43
4	Wnioski i podsumowanie	49
A	Użyte oznaczenia	51
B	Kluczowe fragmenty kodu źródłowego programu	53

Spis rysunków

1.1	Struktura wiekowa populacji Szwecji oraz dopasowanie krzywej wykładniczej	12
1.2	Przykładowy genom dla osobnika w wieku $a = 4$	14
1.3	Śmiertelność w funkcji wieku dla modelu logistycznego i modelu Penny	17
1.4	Wyniki symulacji dla standardowego modelu Penny	18
1.5	Model genomu zastosowany w symulacji – fragment przykładowego genomu dla osobnika w wieku $a = 4$	20
2.1	Zależność śmiertelności od wieku dla populacji USA (1959, źródło [1]), oraz dopasowana krzywa $\alpha \cdot \exp(\beta a)$ dla przedziału wiekowego $a \in (30, 90)$	22
2.2	Wyniki przykładowej symulacji dla $P = 0$ i $P = 0.1$ – demonstracja niepożądanego efektu.	24
2.3	Porównanie wyników symulacji (śmiertelność) z danymi demograficznymi	26
2.4	Porównanie zależności rozmiaru populacji od wieku dla danych demograficznych i wyników symulacji	27
3.1	Porównanie $n(t)$ dla różnych warunków początkowych	29
3.2	Funkcja $m(k)$ dla różnych kroków czasowych. Populacja startowa ($t = 0$) posiada genomy z prawdopodobieństwem wystąpienia mutacji $P_m = 0.5$ dla każdego bitu	30
3.3	Funkcja $m(k)$ dla różnych kroków czasowych. Populacja startowa ($t = 0$) posiada genomy bez mutacji	31
3.4	Zależność śmiertelności od wieku dla różnych sposobów zliczania (rezultat przykładowej symulacji).	33

3.5	Typowe wyniki symulacji dla modelu Penny. Górny wykres przedstawia zależność $n(a)$ (uśrednioną), wraz ze słupkami błędów (odchylenie standardowe), które widoczne są jedynie dla wartości a z górnej części prezentowanego przedziału. Na dolnym wykresie znajduje się zależność stosunku odchylenia standardowego wielkości n do niej samej, w zależności od a	34
3.6	Liczba linii genetycznych w funkcji liczby kroków symulacji. Standardowy model Penny	36
3.7	Struktura populacji – wynik symulacji, uzyskany po 500 ('□') i 20000 ('×') krokach czasowych.	36
3.8	Porównanie ewolucji $G(t)$ dla modelu seksualnego i aseksualnego.	37
3.9	liczba linii genetycznych (znormalizowana) w funkcji czasu, oraz funkcja rozrzutu; współczynnik korelacji pomiędzy funkcją rozrzutu a liczbą linii genetycznych w zależności od dolnej granicy przedziału czasowego.	40
3.10	Zależność funkcji rozrzutu od liczby linii genetycznych w populacji.	42
3.11	Porównanie struktury populacji – wyniki symulacji dla modelu płciowego z preferencją doboru partnera dla różnych wartości parametru r	44
3.12	Porównanie wyników symulacji ($n(t)$ i $G(t)$) dla modelu płciowego z preferencją doboru partnera dla różnych wartości parametru r	45
3.13	Porównanie struktury populacji – wyniki symulacji dla modelu płciowego z preferencją doboru partnera dla różnych wartości parametru l	47
3.14	Porównanie zależności $n(t)$ i $G(t)$ dla modelu płciowego z preferencją doboru partnera, dla różnych wartości parametru $l \in \{0, 8, 16\}$.	48

Rozdział 1

Wstęp

1.1 Cele pracy

Niniejsza praca przedstawia i omawia wyniki symulacji komputerowych Monte-Carlo dynamiki rozwoju populacji biologicznej, na bazie modelu Penny. Model ten zaprezentowany po raz pierwszy w pracy “*A Bit-String Model for Biological Ageing*” [2], zakłada istotny wpływ efektu kumulowania się mutacji (mających ujemny wpływ na organizm, choć nie wyklucza się mutacji dających efekt odwrotny) podczas przekazywania genomu kolejnym pokoleniom, na strukturę populacji¹.

Pomimo tego, iż model Penny dotyczy starzenia biologicznego oraz ewolucji populacji, jest on modelem w którym dominują aspekty fizyczne. Zastosowana metoda obliczeń – Monte-Carlo – pozwala na użycie do opisu narzędzi znanych z mechaniki statystycznej. Stąd też, nie rościmy sobie prawa do opisu takich rzeczy jak przyczyny mutacji genetycznych, czy też szeregu innych zjawisk o podłożu biologicznym (jak choćby biologiczne podstawy rozmnażania i dziedziczenia). W ramach przyjętego modelu, zajmiemy się opisem tych aspektów, które dają się z łatwością ująć w języku fizyki.

Cele pracy są następujące:

- ocena dynamiki zaniku liczby rodzin genetycznych w zależności od przyjętego modelu: wykażemy zasadniczą różnicę pomiędzy charakterem zaniku rodzin genetycznych (“efekt Ewy”) dla modelu seksualnego i aseksualnego. Omówimy również wpływ wspomnianego efektu na pozostałe zależności charakteryzujące populację;
- ocena wpływu selektywnego doboru partnera na strukturę populacji: przed-

¹przez strukturę populacji rozumiemy wszelkie jej parametry statystyczne, jak np. śmiertelność, rozkład prawdopodobieństwa mutacji, zależność liczby osobników od wieku itp.

stawimy modyfikację seksualnego modelu Penny, która pozwala na określenie preferencji doboru partnerów. Pokażemy również zależność pomiędzy sposobem doboru partnera, a tempem zanikania rodzin genetycznych.

- analiza słuszności prawa Gompertza dla różnych mechanizmów śmierci: omówimy wpływ poszczególnych mechanizmów na zależność śmiertelności od wieku, oraz przeprowadzimy dyskusję na temat wpływu czynnika Verhulsta na strukturę populacji.

Tematyka ta, zostanie przedstawiona i omówiona w rozdziale 3.

W celu zbadania wiarygodności naszej implementacji modelu, a przez to i otrzymanych wyników, w podrozdziale 1.4 odtworzymy powszechnie znane wyniki osiągnięte przy zastosowaniu modelu Penny – między innymi takie jak:

- zależność liczby osobników w populacji od wieku,
- zależność śmiertelności od wieku,
- zestawienie otrzymanej śmiertelności z danymi demograficznymi.

1.2 Możliwe przyczyny biologicznego starzenia

Istnieje szereg teorii mających na celu wyjaśnienie przyczyn biologicznego starzenia się [3]. Można podzielić je na dwie grupy:

- teorie bazujące na negatywnym wpływie środowiska jako głównej przyczynie starzenia;
- teorie opierające się na “zaprogramowanej śmierci”.

Do pierwszej grupy można zaliczyć: teorię zużywania się (organizm starzeje się ponieważ kumulują się w nim uszkodzenia tkanek i organów), akumulacji odpadów (komórki nie są w stanie odpowiednio szybko usuwać nagromadzonych odpadów), “usieczowania” (postępujące zmiany w białku – kolagenie), wolnych rodników (grupa związków o tej nazwie wchodzi w reakcję z DNA uszkadzając go), systemu immunologicznego (obniżona zdolność do wytwarzania przeciwciał, oraz wytwarzanie ich dla białek własnego organizmu), czy też teorii nagromadzenia się mutacji somatycznych (nabytych przez komórki organizmu w trakcie życia).

Drugą grupę stanowią między innymi teorie takie jak: teoria neurohormonalna (ograniczenie np. zdolności rozrodczej powodowane określonym zachowaniem układu hormonalnego), teoria “antagonistycznej plejotropii” (geny które wpływają korzystnie na organizm w wieku rozrodczym, mają zły wpływ w późniejszych latach życia), oraz teoria akumulacji mutacji.

Teoria akumulacji mutacji będąca podstawą modelu Penny, zakłada iż starzenie się organizmu powodowane jest przez szkodliwe mutacje w genomie osobnika, które są dziedziczone od przodka (lub przodków w przypadku rozmnażania płciowego). Istotne znaczenie ma fakt, iż podczas przekazywania genomu potomkowi mogą wystąpić błędy w powielaniu DNA, co powoduje dodatkowe (na ogół szkodliwe) mutacje. Wpływ szkodliwych mutacji na organizm jest jednak opóźniony – nie ujawniają się zaraz po urodzeniu, lecz są powoli “odsłaniane” w kolejnych latach życia (ujawniając się np. jako choroby). Gdy liczba owych “odsłoniętych” mutacji przekroczy wartość krytyczną – osobnik umiera. Teoria ta nie uwzględnia roli mutacji somatycznych.

Publikacje dotyczące wymienionych tu teorii dotyczących możliwych przyczyn starzenia, (jak między innymi [4], opisująca wpływ wolnych rodników na procesy starzenia) koncentrują się na ogół, na bezpośrednich efektach niż wynikach ilościowych. Nie mniej jednak, autorzy artykułu [5] odtworzyli eksponencyjną zależność śmiertelności od wieku na bazie teorii antagonistycznej plejotropii.

1.3 Symulacje Monte-Carlo

Nazwa Monte-Carlo obejmuje grupę metod numerycznych służących do symulacji zjawisk fizycznych lub też obliczeń matematycznych. Spośród innych metod, Monte-Carlo wyróżniają się tym, że w obliczeniach używany jest ciąg liczb losowych (lub pseudolosowych). Najbardziej klasycznym przykładem ich użycia są metody Monte-Carlo liczenia całek oznaczonych (na ogół wielowymiarowych). Dla funkcji jednej zmiennej $f(x) : A \subset \mathbb{R} \rightarrow B \subset \mathbb{R}$, metoda taka polega ona na wylosowaniu N punktów o współrzędnych (x_i, y_i) (gdzie $i \in \{1 \dots N\}$), takich że:

$$x_i \in A \quad \text{ i } \quad y_i \in B, \quad (1.1)$$

oraz wylosowane próbki x_i i y_i mają rozkłady jednorodne na przedziale, odpowiednio A i B . Oszacowanie wartości całki oznaczonej na przedziale B otrzymujemy obliczając stosunek liczby punktów dla których rzędna jest większa od wartości funkcji dla ich odciętej, do liczby punktów dla których jest ona mniejsza (przez co dostajemy oszacowanie jaką część prostokąta o wymiarach $A \times B$ zajmuje pole “pod funkcją”). Szacowany błąd tak wyliczonej całki wynosi $1/\sqrt{N}$.

Użycie ciągu liczb pseudolosowych w obliczeniach daje tę przewagę nad ciągiem prawdziwie losowym (poza oczywiście znacznie prostszą metodą uzyskania tego pierwszego), iż wyniki tak przeprowadzonych obliczeń są powtarzalne. Dlatego też najczęściej mamy do czynienia z obliczeniami przy użyciu ciągów liczb uzyskanych z generatorów komputerowych, z których najbardziej popularne są generatory kongruentne [6].

Do obliczeń przeprowadzonych na potrzeby tej pracy, został wykorzystany generator liczb pseudolosowych Parka-Millera [7]. Ciąg liczb pseudolosowych został użyty do wygenerowania populacji początkowej, do wyznaczenia poszczególnych zdarzeń śmierci osobników, (zgodnie z prawdopodobieństwem zadany czynnikiem Verhulsta) oraz ich narodzin (opisano w następnym rozdziale).

1.4 Wybrane modele rozwoju populacji

1.4.1 Model logistyczny

Model logistyczny jest często postrzegany jako uproszczenie modelu Penny. Umożliwia on spojrzenie na jeden z jego aspektów – równowagę pomiędzy liczbą narodzin osobników danej populacji i liczbą przypadków śmierci, która jest tu (w przypadku modelu logistycznego) powodowana wyłącznie przez ograniczone zasoby środowiska.

W jednym kroku czasowym, śmierć osobnika następuje z prawdopodobieństwem $\frac{n}{N}$, gdzie $n \in \mathbb{N}$ jest to aktualna liczba osobników w rozpatrywanym środowisku, zaś $N \in \mathbb{N}$ jest parametrem modelu – pojemnością środowiska. Prawdopodobieństwo to nazywane jest często czynnikiem Verhulsta.

Drugim parametrem modelu jest zdolność rozrodcza pojedynczego osobnika: $B \in \mathbb{R}$. Osobnik który przeżył wstępny etap eliminacji z powodu czynnika Verhulsta, wydaje potomstwo w liczbie $\lfloor B \rfloor$ (największa liczba całkowita, nie większa niż B), oraz dodatkowo jednego potomka z prawdopodobieństwem $P_B = B - \lfloor B \rfloor$.

Zatem ewolucja układu przebiega następująco – w każdym kroku czasowym t :

- każdy osobnik jest poddany eliminacji i jest usunięty z prawdopodobieństwem $P_d = \frac{n(t)}{N}$;
- osobnik który przeżył wydaje potomstwo w liczbie odpowiadającej parametrowi B ;
- wszystkie nowo narodzone osobniki, oraz te które przeżyły, stanowią populację wejściową w następnym kroku czasowym $t + 1$.

Powyższy schemat pozwala na prześledzenie ewolucji populacji. Liczba osobników które przeżyją pierwszy etap, wynosi:

$$n_1 = n(t) - \frac{n(t)}{N} \cdot n(t) = n(t) \cdot \left(1 - \frac{n(t)}{N}\right). \quad (1.2)$$

W drugim etapie – rozmnażania, populacja n_1 wzrasta o nowonarodzone osobniki $B \cdot n_1$, stąd populacja w kroku czasowym $t + 1$:

$$n(t+1) = n(t) \left(1 - \frac{n(t)}{N}\right) + B \cdot n(t) \cdot \left(1 - \frac{n(t)}{N}\right). \quad (1.3)$$

Podstawiając następnie $x(t) = n(t)/N$, oraz $k = B + 1$, otrzymamy równanie:

$$x(t+1) = k \cdot x(t)(1 - x(t)), \quad (1.4)$$

powszechnie znane jako równanie logistyczne [8].

Dla początkowych kroków czasowych symulacji, rezultaty będą zależały od od stanu układu w kroku $t_0 = 0$ (zakładamy niezerowy i mniejszy od pojemności środowiska, rozmiar populacji początkowej). Dlatego też ocena wyników ma sens dopiero wówczas, gdy badany układ osiągnie stan stacjonarny. Rozumiemy przez to stan, w którym wartości oczekiwane wszystkich wielkości obserwowalnych danego układu nie zmieniają się w czasie. Na potrzeby rozważań dotyczących naszego układu, weźmiemy na razie pod uwagę jedynie rozmiar populacji. Przyjmiemy iż układ osiąga stan stacjonarny dla takich t , dla których uśredniony po wielu krokach rozmiar populacji pozostaje w przybliżeniu stały, czyli:

$$\frac{1}{t'_1 - t_1} \sum_{t=t_1}^{t'_1} n(t) \approx \frac{1}{t'_2 - t_2} \sum_{t=t_2}^{t'_2} n(t), \quad (1.5)$$

gdzie $t'_i \gg t_i$.

Dla takiego stanu wyznaczymy rozmiar populacji. Sprowadza się to do wyznaczenia punktu stałego odwzorowania (1.4), czyli takiego x^* , że:

$$x^* = kx^*(1 - x^*). \quad (1.6)$$

Po przekształceniu powyższego równania dostajemy:

$$x^*(k(1 - x^*) - 1) = 0, \quad (1.7)$$

skąd otrzymujemy dwa rozwiązania – jedno zerowe $x_0^* = 0$, oraz nietrywialne:

$$x^* = 1 - \frac{1}{k} = \frac{B}{B+1}. \quad (1.8)$$

Zatem liczba osobników w stanie stacjonarnym wynosi:

$$n_{st} = \frac{NB}{B+1}. \quad (1.9)$$

W celu dalszej analizy modelu, zdefiniujemy dwie wielkości charakteryzujące populację – współczynnik przetrwania oraz śmiertelność. Współczynnikiem przetrwania dla osobników w wieku a , nazywamy wielkość:

$$s(a) = \frac{n(a+1)}{n(a)}, \quad (1.10)$$

gdzie $n(a)$ jest liczbą osobników w wieku a , przy czym gdy nie jest spełnione (1.5) wówczas $n(a) = n(t, a)$. Śmiertelnością nazywamy wielkość:

$$q(a) = 1 - s(a) = \frac{n(a) - n(a+1)}{n(a)}. \quad (1.11)$$

Aby wyznaczyć obie wielkości, wyliczymy ile (średnio) osobników w wieku a w kroku czasowym t , przetrwa do następnego kroku $t+1$:

$$n(t+1, a+1) = n(a) - \frac{n}{N}n(a) = n(a) \left(1 - \frac{n}{N}\right) = n(a) \left(1 - \frac{B}{B+1}\right), \quad (1.12)$$

gdzie $n = n(t)$ jest rozmiarem populacji w kroku czasowym t . Ponieważ rozpatrujemy nasz układ w stanie stacjonarnym, możemy przyjąć że:

$$n(t+1, a+1) = n(t, a+1). \quad (1.13)$$

Zatem współczynnik przetrwania wyniesie:

$$s = \frac{n(t, a+1)}{n(t, a)} = \frac{1}{B+1}, \quad (1.14)$$

zaś śmiertelność:

$$q = 1 - s = \frac{B}{B+1}. \quad (1.15)$$

Jak widać współczynniki s i q nie zależą od wieku a , co jest prostą konsekwencją faktu, iż prawdopodobieństwo śmierci osobnika w pojedynczym kroku czasowym nie zależy w naszym modelu od jego wieku, a jedynie od całkowitego rozmiaru populacji. Ponieważ obserwowana śmiertelność w rzeczywistych populacjach (dane demograficzne – [1], [9], [10]), silnie zależy od wieku, fakt ten dyskryminuje model logistyczny jako przydatny do symulowania struktury tychże populacji. Pomimo tego charakter zależności $n(a)$ pozostaje w zgodzie z danymi doświadczalnymi, co wykażemy poniżej.

Weźmy pod uwagę liczbę osobników w wieku $a+1$:

$$n(t+1, a+1) = n(a) \left(\frac{1}{B+1}\right) = n(t, a+1). \quad (1.16)$$

Dla poszczególnych a , dostaniemy:

$$\begin{aligned} n(1) &= n(0) \left(\frac{1}{B+1}\right) \\ n(2) &= n(1) \left(\frac{1}{B+1}\right) = n(0) \left(\frac{1}{B+1}\right)^2 \\ &\vdots \\ n(a) &= n(0) \left(\frac{1}{B+1}\right)^a = n(0) \cdot s^a. \end{aligned} \quad (1.17)$$

Do pełnej znajomości $n(a)$ brakuje nam jeszcze $n(0)$. Aby je wyznaczyć, wystarczy zauważyć, iż:

$$n = \sum_{a=0}^{\infty} n(a) = \sum_{a=0}^{\infty} n(0)s^a. \quad (1.18)$$

Ponieważ $s \in (0, 1)$:

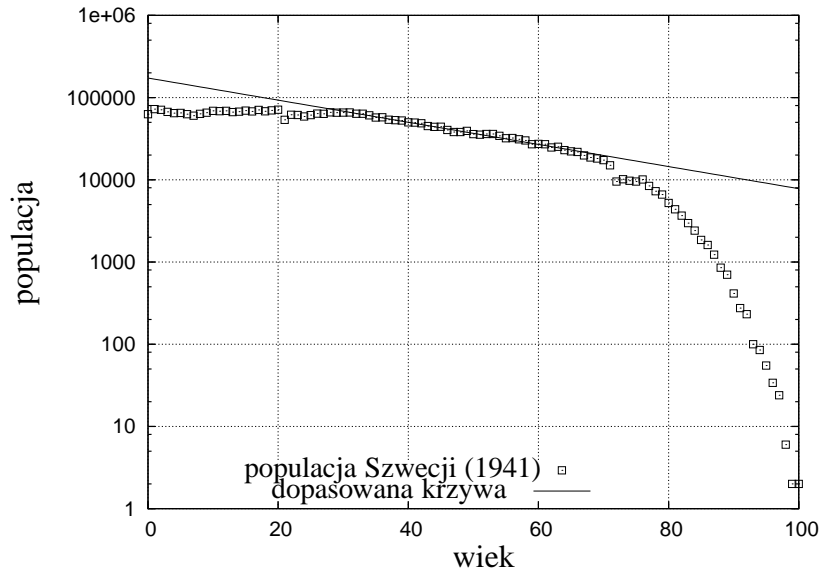
$$n = n(0) \frac{1}{1-s}. \quad (1.19)$$

Tak więc:

$$n(0) = N \frac{n}{N} \left(1 - \frac{1}{B+1}\right) = N \frac{B}{B+1} \cdot \frac{B}{B+1} = Nq^2. \quad (1.20)$$

Stąd otrzymujemy zależność:

$$n(a) = Nq^2 s^a = N \cdot \left(\frac{B}{B+1}\right)^2 \cdot \left(\frac{1}{B+1}\right)^a \quad (1.21)$$



Rysunek 1.1: Struktura wiekowa populacji Szwecji z roku 1941 (źródło [1]), oraz dopasowanie krzywej wykładniczej $n_0 \cdot s^a$ dla obszaru wiekowego $a \in (30, 70)$

Zależność wykładnicza $n(a)$ jest również obserwowana w przypadku rzeczywistych populacji, co zaprezentowano na rysunku (1.1). Wybór tak odległych lat nie jest przypadkowy. Powojenne dane demograficzne noszą silne znamiona II wojny światowej, której efekty widać w strukturach obecnych populacji, szczególnie europejskich. Jak można zaobserwować na wykresie – $n(a)$ ma charakter

wykładniczy jedynie w środkowym przedziale wiekowym (30, 70), co jest jednak sukcesem biorąc pod uwagę prostotę prezentowanego modelu.

Istotną kwestią w przypadku prezentowanego modelu, jest również to czy otrzymany punkt stały $x^* = \frac{B}{B+1}$ jest punktem przyciągającym, czyli ciąg:

$$x_0, x_1, \dots \quad (1.22)$$

jest zbieżny do x^* w pewnym jego otoczeniu.

Jak zostało to przedstawione w [8], zbieżność takiego ciągu, generowanego przy pomocy dowolnego równania iteracyjnego:

$$x_{n+1} = F(x_n), \quad (1.23)$$

jest gwarantowana gdy spełniony jest warunek:

$$|F'(x)| < 1. \quad (1.24)$$

Dla naszego równania logistycznego:

$$F(x) = kx(1-x), \quad (1.25)$$

zatem pochodna wynosi:

$$F'(x) = -2kx + k. \quad (1.26)$$

Jeśli teraz nałożymy na to warunek 1.24, otrzymamy:

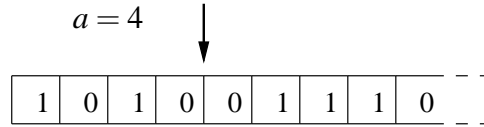
$$k \in (1, 3) \quad (1.27)$$

Zatem punkt stały i przyciągający w przypadku modelu logistycznego osiągniemy przy $B \in (0, 2)$. Wszystkie prezentowane w pracy wyniki, są wynikami dla parametrów początkowych uwzględniających ten warunek.

Niewątpliwą zaletą modelu logistycznego jest fakt, iż potrafimy otrzymać jego analityczne rozwiązania. Jest to pomocne szczególnie podczas analizy rezultatów symulacji na bazie modelu Penny, który może być postrzegany jako jego uogólnienie. Brak zależności śmiertelności od wieku, oraz między innymi brak cech indywidualnych poszczególnych osobników są wadami na tyle poważnymi, że skłaniają nas do użycia modeli lepiej odtwarzających rzeczywistość.

1.4.2 Model Penny

Model Penny opiera się na teorii akumulacji mutacji opisanej w poprzednim rozdziale. Pierwszoplanową rolę odgrywa w nim dziedziczenie przez potomków cech rodziców. Pozwala to na spojrzenie na populację z ewolucyjnego punktu widzenia, oraz na uwzględnienie wpływu czynników genetycznych na jej strukturę.



Rysunek 1.2: Przykładowy genom dla osobnika w wieku $a = 4$ (liczba aktywnych mutacji $\mu(4) = 2$).

Podobnie jak w przypadku modelu logistycznego, czynnik Verhulsta pomaga utrzymać skończony rozmiar populacji, lecz w przeciwieństwie do niego nie jest jedynym czynnikiem wpływającym na śmiertelność. Każdy osobnik posiada genom g składający się z ciągu bitów o stałej długości L (rys. 1.2).

Na każdej pozycji genomu '1' oznacza obecność szkodliwej mutacji, zaś '0' jej brak. Genom jest w całości dziedziczony od przodka (model aseksualny) w momencie urodzenia. Dodatkowo jest on wówczas wzbogacony o $M \in \mathbb{R}$ mutacji (a ściślej o $\lfloor M \rfloor$ mutacji, oraz dodatkowo jedną mutację z prawdopodobieństwem $P_M = M - \lfloor M \rfloor$). Model uwzględnia również możliwość występowania mutacji naprawczych (występujących znacznie rzadziej) – ich występowanie kontroluje parametr P (analogicznie jak parametr M). W każdym kroku czasowym t , zwiększany jest o 1 wiek a osobnika, oraz odsłaniane są kolejne bity genomu. Tak więc w wieku a wyeksponowanych jest tylko pierwszych a bitów (rys. 1.2). Liczbę wszystkich "jedynek" w genomie obecnych na pozycjach od pierwszej do a -tej, nazywamy liczbą mutacji aktywnych $\mu(a)$ (jak widać jest ona określona dla każdego osobnika już w momencie urodzenia). Gdy liczba ta osiągnie zadany próg $T \in \mathbb{N}$ (parametr modelu), osobnik umiera.

W modelu Penny ewolucja układu przebiega następująco – w pojedynczym kroku czasowym t :

- każdy osobnik zmienia swój wiek z a na $a + 1$;
- każdy osobnik jest poddany eliminacji z przyczyn:
 - genetycznych – gdy liczba mutacji aktywnych $\mu(a)$ osiąga próg T ;
 - środowiskowych – analogicznie jak w modelu logistycznym o śmierci decyduje czynnik Verhulsta, tzn. osobnik umiera z prawdopodobieństwem $P_d = \frac{n(t)}{N}$;
 - pozostałych – gdy osobnik osiągnie maksymalny możliwy wiek równy długości genomu: $a = L$ (w stanie stacjonarnym sytuacja ta praktycznie nie występuje, nie wnosi więc ona istotnego wkładu do całkowitej statystyki śmiertelności);

- każdy osobnik który przeżył oraz osiągnął minimalny wiek rozrodczy $R \in \mathbb{N}$ – wydaje potomstwo w liczbie $\lfloor B \rfloor$ oraz jednego potomka z prawdopodobieństwem $P_B = B - \lfloor B \rfloor$;
- każdy nowo narodzony osobnik dziedziczy genom od przodka, z uwzględnieniem dodatkowych M mutacji szkodliwych i P mutacji pozytywnych, umieszczanych w losowych miejscach w genomie. Każda mutacja negatywna (pozytywna) oznacza ustawienie bitu na danej pozycji na '1' ('0'), niezależnie od jego poprzedniej wartości;
- wszystkie narodzone osobniki, oraz te które przeżyły stanowią populację wejściową w następnym kroku czasowym $t + 1$.

Populację startową wybieramy albo losując mutacje dla poszczególnych osobników z zadaniem prawdopodobieństwem, albo też zostawiając wszystkie pozycje w genomach ustawione na '0'. Obie metody dają na ogół zbliżone (w sensie wartości uśrednionych) rezultaty po odpowiednio długim czasie symulacji.

Opisany model posiada zatem sześć parametrów wejściowych (N, B, R, M, P, T) , nazywanych kolejno:

- N – pojemność środowiska;
- B – zdolność rozrodcza;
- R – minimalny wiek reprodukcji;
- M – współczynnik mutacji;
- P – współczynnik mutacji pozytywnych;
- T – próg mutacji aktywnych.

Przedstawiony model, możemy łatwo sprowadzić do modelu logistycznego kładąc $M = 0$ (przy założeniu, iż populacja początkowa nie posiada żadnych mutacji), lub też przechodząc w granicy z $T \rightarrow \infty$ (opisano w [11]).

Uściślając wprowadzone pojęcia, genom i -tego osobnika określimy jako:

$$g_i(k) \in \{0, 1\}, \quad \text{gdzie } k \in \{1, 2, 3, \dots, L\}, \quad i \in \{1, 2, \dots, n\}, \quad (1.28)$$

zaś liczbę jego aktywnych mutacji $\mu_i(a)$ jako:

$$\mu_i(a) = \begin{cases} \sum_{k=1}^a g_i(k) & \text{dla } a > 0 \\ 0 & \text{dla } a = 0 \end{cases} \quad (1.29)$$

Populację $n(t)$ w kroku czasowym t uzyskaną w wyniku symulacji, możemy scharakteryzować nieco bardziej dokładnie niż to miało miejsce w przypadku modelu logistycznego – jako sumę podpopulacji $n(t, a, \mu)$. Stąd całkowity rozmiar populacji otrzymamy jako:

$$n(t) = \sum_{a, \mu} n(t, a, \mu). \quad (1.30)$$

Ponieważ najczęściej interesuje nas jak to, zachowuje się nasz układ w stanie stacjonarnym, rozpatrujemy n po odpowiednio długim czasie:

$$n(a, \mu) = n(t \rightarrow \infty, a, \mu), \quad (1.31)$$

zatem liczba osobników w wieku a wynosi:

$$n(a) = \sum_{\mu} n(a, \mu). \quad (1.32)$$

Jak zauważyliśmy w poprzednim rozdziale, istotną wadą modelu logistycznego jest fakt iż $q(a) = \text{const}$. Model Penny pozwala na uzyskanie wyników znacznie bardziej zbliżonych do rzeczywistych. Rysunek 1.3 przedstawia porównanie zależności śmiertelności od wieku dla obu modeli (wyniki symulacji). Dla modelu logistycznego przedstawiono całkowitą śmiertelność, zaś dla modelu Penny jedynie śmiertelność liczoną dla przypadków śmierci z przyczyn genetycznych (przekroczenie progu aktywnych mutacji), co powoduje, iż dla wieku $a < T$ otrzymujemy śmiertelność zerową. Zależność wykładnicza zgodna jest z empirycznym prawem Gompertza [12] dotyczącym populacji ludzkiej (opisano w rozdziale 3.2).

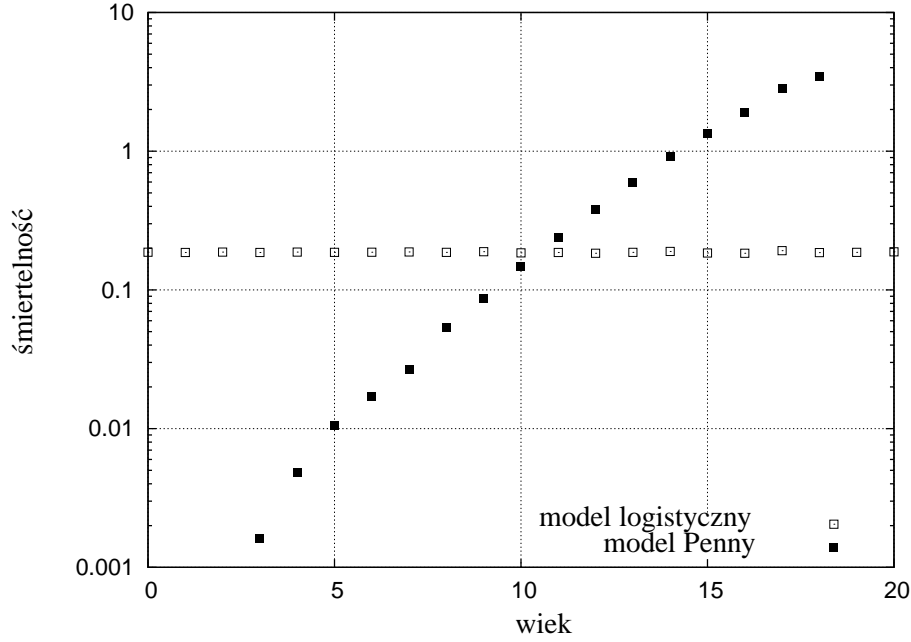
Z ewolucyjnego punktu widzenia istotny jest nie tyle fakt, czy pojedynczy osobnik przeżyje i jaka będzie długość jego życia, lecz to czy zdoła przetrwać konkretny genotyp. Aby to było możliwe, “nosiciel” danego genomu musi mieć możliwość jego zreplikowania (możliwość rozmnożenia się). Ponieważ w naszym modelu minimalny wiek reprodukcji wynosi R , replikowane będą tylko genomy spełniające warunek:

$$\sum_{k=1}^R g_i(k) < T \quad (1.33)$$

Fakt ten znajduje odzwierciedlenie w funkcji rozkładu mutacji $m(k)$, określonej jako:

$$m(k) = \frac{1}{n} \sum_{i=1}^n g_i(k), \quad \text{gdzie } k \in (1, 2, \dots, L) \quad (1.34)$$

Funkcję tę (wynik symulacji) przedstawia prawy, dolny wykres na rysunku 1.4. Bity na pozycjach początkowych odsłaniane w początkowej fazie życia, mają średnie wartości znacznie mniejsze niż bity na pozycjach dalszych (gdzie funkcja rozkładu mutacji osiąga wartości $m(k) \approx 1$). Jest to efekt równowagi pomiędzy presją

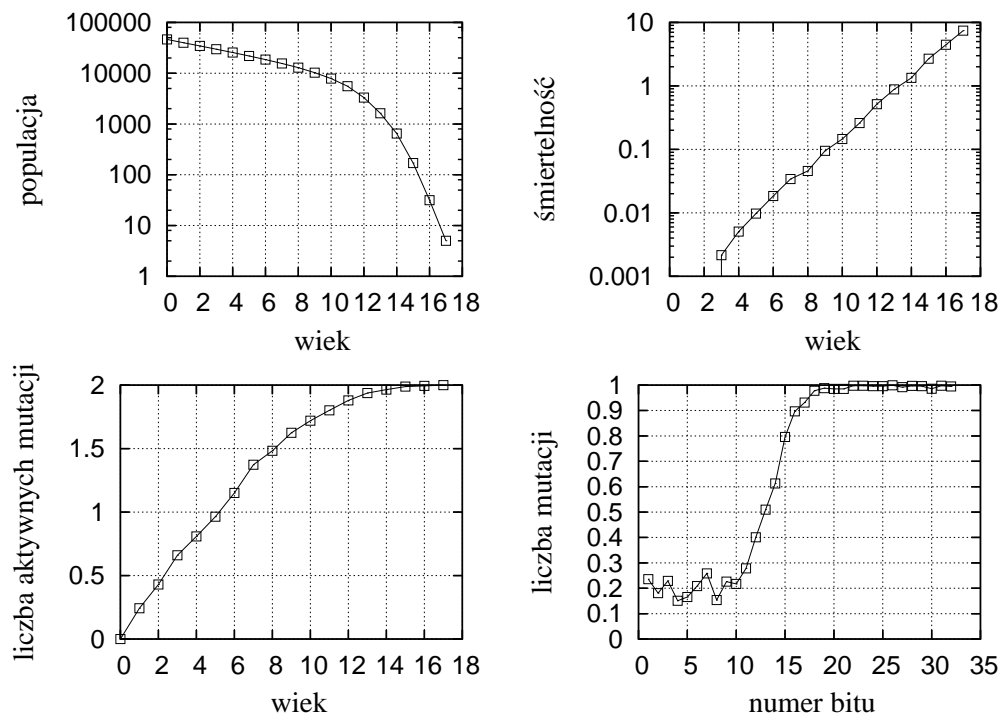


Rysunek 1.3: Śmiertelność w funkcji wieku dla modelu logistycznego: $(N, B, R, M, P, T) = (10^6, 0.8, 7, 0, 0, 3)$ i dla modelu Penny: $(N, B, R, M, P, T) = (10^6, 0.8, 7, 1, 0.1, 3)$

mutacyjną, a mechanizmem ewolucyjnym, który eliminuje genomy nie stwarzające szansy na powielenie (nie spełniające warunku (1.33)).

Lewy dolny wykres na rysunku 1.4 przedstawia zależność $\mu_i(a)$ uśrednioną po wszystkich osobnikach – $\bar{\mu}(a)$. Ponieważ próg mutacji w prezentowanej symulacji wynosi $T = 3$, maksymalna osiągnięta wartość wynosi $\mu_{max} = T - 1 = 2$. Prawy górny wykres przedstawia prezentowaną wcześniej zależność śmiertelności z przyczyn genetycznych od wieku osobnika, zaś lewy górny – rozmiar populacji w zależności od wieku. Zależność $n(a)$ podobnie jak $q(a)$, odbiega od wyników otrzymanych na bazie modelu logistycznego, dla którego dostaliśmy $n(a) \sim s^a$. Uwagę zwraca “załamanie” krzywej w okolicach $a = 12$ występujące w danych demograficznych, jak choćby na prezentowanym wcześniej wykresie dla populacji Szwecji 1.1

Model Penny może zostać w prosty sposób poszerzony o dodatkowe własności jak na przykład: uwzględnienie płci osobników (opisane w następnym podrozdziale oraz w [13], [14], [15], [16]), biologiczna skala czasowa [17], uwzględnienie rozmieszczenia osobników w przestrzeni [18], zmienny próg T [19], czy też migracje [20].



Rysunek 1.4: Wyniki symulacji – standardowy model Penny $(N, B, R, M, P, T) = (2 \cdot 10^6, 0.8, 7, 0.9, 0, 3)$ (wyniki uśrednione po 20 krokach w stanie stacjonarnym)

1.4.3 Model Penny z uwzględnieniem płci osobników

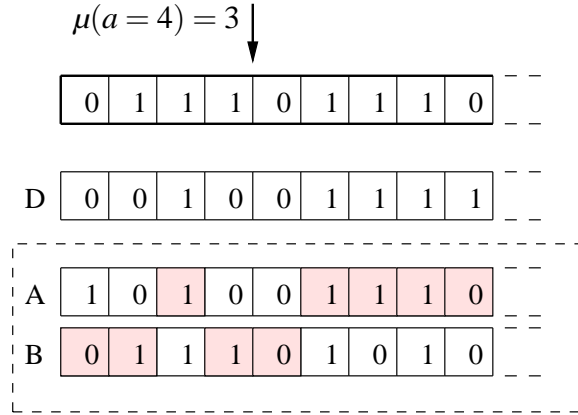
Standardowy model Penny, zakłada bardzo prosty sposób rozmnażania organizmów, “pączkują” one przekazując swój genom bezpośrednio potomkom. Mimo iż model ten daje wyniki zgodne z danymi doświadczalnymi, zasadne wydaje się pytanie – jak sposób rozmnażania wpływa na strukturę populacji. Zanim przedstawimy sposób implementacji modelu płciowego, opiszemy krótko jego biologiczne podstawy [21].

Materiał genetyczny w organizmach żywych zlokalizowany jest w strukturach nazywanych chromosomami. Chromosomy te (u części z organizmów zwanych *diploidalnymi*, w tym u człowieka) można pogrupować w pary o takim samym zestawie informacji genetycznej². Chromosomy pochodzące z jednej pary nazywane są *homologicznymi*. Chromosomy homologiczne kodują te same cechy lecz nie są identyczne, efekt ich działania zależy od tego który z dwóch genów kodujących tę samą cechę organizmu, okaże się dominujący. Istotą przekazywania genów potomkom w rozmnażaniu płciowym, jest proces *rekombinacji* (zwany też *crossing-over*). Polega on na tym iż dwa homologiczne chromosomy “przełamują” się w losowym miejscu i następnie każdy z nich łączy się z fragmentem pochodzącym od drugiego chromosomu. W ten sposób powstaje nowy układ genów w chromosomach. Komórki rozrodcze (*gamety*) posiadają po jednym zestawie chromosomów powstałych po rekombinacji – są *haploidalne*. Tak więc pojedynczy organizm produkuje różne rodzaje gamet. Zygota powstaje w wyniku połączenia się dwóch gamet – żeńskiej i męskiej i posiada ponownie dwa zestawy chromosomów – jest diploidalna tak samo jak organizmy rodziców.

W naszej modyfikacji modelu Penny, organizmy posiadają tylko jedną parę homologicznych chromosomów (w przeciwieństwie do organizmu człowieka który posiada ich 23) nazwanych A i B – są to dwa ciągi bitów o długości L każdy. Organizmy zostały podzielone na żeńskie i męskie. W każdym kroku czasowym podczas rozmnażania, następuje losowy dobór partnera płci przeciwnej (dotyczy to osobników które osiągnęły minimalny wiek reprodukcji R). Każda para wydaje potomstwo w liczbie odpowiadającej parametrowi B (analogicznie jak dla modelu aseksualnego). Proces narodzin potomka przebiega następująco:

- dla każdego z rodziców:
 - losowana jest z rozkładem jednorodnym liczba α z przedziału $(0, L)$;
 - bity na pozycjach od 0 do α z chromosomu A, oraz bity na pozycjach od $\alpha + 1$ do L z chromosomu B, wchodzi w skład gamety pierwszej (ciąg bitów o długości L);

²wyjątek stanowią chromosomy X i Y u mężczyzny lecz nie uwzględniamy tego faktu w naszej prostej modyfikacji standardowego modelu Penny



Rysunek 1.5: Model genomu zastosowany w symulacji – fragment przykładowego genomu dla osobnika w wieku $a = 4$.

- bity na pozycjach od 0 do α z chromosomu B, oraz od $\alpha + 1$ do L z chromosomu A, wchodzi w skład gamety drugiej (gamety posiadają więc po jednym chromosomie);
- zygota powstaje z jednej (wylosowanej z prawdopodobieństwem $1/2$) gamety męskiej i jednej (również wylosowanej) gamety żeńskiej, jest więc diploidalna tak jak genomy rodziców;
- w chromosomach potomka umieszczane są mutacje w liczbie odpowiadającej parametrowi M (oraz odpowiednio P). Losowane jest zarówno miejsce wystąpienia dodatkowej mutacji (analogicznie jak w modelu aseksualnym) jak i chromosom w którym mutacja ma wystąpić (z prawdopodobieństwem $1/2$)
- płeć potomka wybierana jest z prawdopodobieństwem $1/2$.

Nieco problematyczny wydaje się sposób “odsłaniania” poszczególnych bitów genomu podczas symulacji w celu obliczenia liczby mutacji aktywnych μ . Sposób ten został schematycznie przedstawiony na rysunku 1.5. Oba chromosomy czytane są jednocześnie, zaś to który bit genomu jest dominujący, określa dodatkowy ciąg bitów D o długości L odpowiadającej długości pojedynczego chromosomu (pomysł zaczerpnięto z książki [13]). Ciąg ten jest losowany na początku symulacji i nie zmienia się w podczas jej przebiegu. Każdy jego bit ustawiony na ‘1’ oznacza że na danej pozycji dominujący jest gen z chromosomu A i on jest brany pod uwagę przy zliczaniu liczby aktywnych mutacji, zaś dla ‘0’ dominujący jest gen z chromosomu B. Wartość genomu na dominującej pozycji k , dla i -tego osobnika, będziemy oznaczali przez $g^*(i, k)$.

Rozdział 2

Przegląd badań nad procesami ewolucji populacji

2.1 Linie genetyczne

W podrozdziale tym przedstawimy problemem spadku różnorodności genomów populacji wraz ze wzrostem czasu t podczas przebiegu symulacji. Zanim zaprezentujemy efekty tego zjawiska i jego ilościową analizę w podrozdziale 3.3, opiszemy krótko jego potencjalne przyczyny. Pierwszą z możliwych przyczyn jest fakt, iż używany przez nas model opisuje populację izolowaną – nie kontaktuje się ona z innymi, które mogły ewoluować w nieco odmiennych warunkach (nie dochodzi więc do wymiany materiału genetycznego). Sytuacja ta jest wydajna dla opisu dynamiki rozwoju małej populacji na wyspie bez kontaktu z resztą świata, niż obecnego stanu demograficznego. Drugą potencjalną przyczyną jest fakt, że środowisko jest stałe w czasie – dany gen który był niekorzystny dla właściciela, pozostanie takim do końca symulacji, w przeciwieństwie do środowiska rzeczywistego. Problem ten został szerzej opisany w [22].

Aby prześledzić proces rozprzestrzeniania się genotypów w środowisku, wprowadza się na ogół modyfikację do modelu, dodatkową cechę dla poszczególnych osobników. Podczas generowania populacji początkowej, każdy osobnik zostaje oznaczony etykietą – liczbą naturalną. Dla populacji początkowej etykiety się nie powtarzają, liczba etykiet odpowiada liczbie osobników. W trakcie kolejnych kroków symulacji, każdy nowonarodzony osobnik dziedziczy etykietę po przodku¹ tak więc liczba etykiet w populacji nie może rosnąć. Zbiór osobników o tej samej etykiecie nazywamy linią genetyczną. Należy zauważyć iż to, że osobniki należą do jednej linii genetycznej, nie oznacza że mają identyczne genomy, gdyż podczas dziedziczenia genomy potomków wzbogacane są o dodatkowe mutacje.

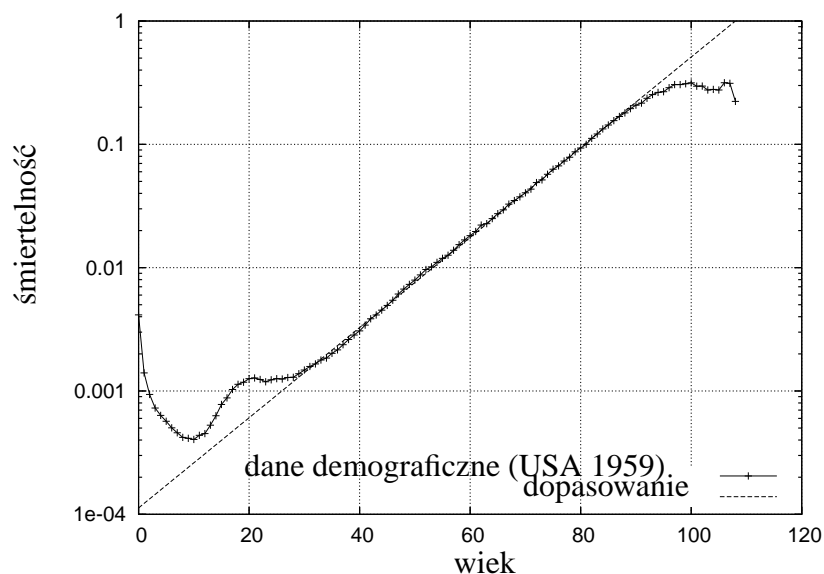
¹W przypadku modelu płciowego, etykieta jest dziedziczona tylko po rodzicu jednej płci.

Wprowadzona cecha osobników, pozwala na zaobserwowanie zjawiska zwanego “efektem Ewy” ([23], [24], [25]). Zjawisko to polega na tym, iż niezależnie od doboru warunków początkowych (dających stabilną, niewymierającą populację), symulacja prowadzi zawsze do stanu, w którym zostaje tylko jedna linia genetyczna w populacji. Efekt ten znany jest zarówno w ramach modelu Penny, jak i poza nim. Dla rzeczywistej populacji ludzkiej, korespondująca ze wspomnianym efektem hipoteza (zwana “hipotezą mitochondrialnej Ewy”), została wysunięta na podstawie badań nad mitochondrialnym DNA, które jest dziedziczone jedynie od matki potomka. Sugeruje ona, iż cała populacja ludzka pochodzi od jednego przodka (w linii żeńskiej), co jest odtwarzalne na bazie modelu Penny (pomimo prostego, bezpłciowego modelu rozmnażania).

Zarówno samo zjawisko jak i dynamika zaniku linii genetycznych, zostaną omówione w rozdziale 3.3

2.2 Zasadnicze wyniki w aseksualnym modelu Penny

2.2.1 Prawo Gompertza



Rysunek 2.1: Zależność śmiertelności od wieku dla populacji USA (1959, źródło [1]), oraz dopasowana krzywa $\alpha \cdot \exp(\beta a)$ dla przedziału wiekowego $a \in (30, 90)$.

Śmiertelność w populacji ludzkiej zależy od wielu czynników, dlatego też trudno jest o jednoznaczny i prosty opis zależności śmiertelności od wieku. Na

statystykę śmiertelności oprócz przyczyn naturalnych, mają też wpływ takie czynniki jak wypadki drogowe, czy też opieka lekarska. Nie mniej jednak, w pewnym zakresie możliwe jest ujęcie zależności śmiertelności od wieku jako danej empirycznym prawem Gomperta [13], [12]. Mówi ono, iż zależność śmiertelności od wieku ma charakter wykładniczy:

$$q(a) = \alpha \cdot \exp(\beta a), \quad (2.1)$$

gdzie a oznacza wiek osobnika, a α i β są stałymi. Aby ocenić zgodność tego prawa z rzeczywistością, zobaczmy jak wygląda przykładowa zależność $q(a)$ dla populacji ludzkiej, przedstawiona na wykresie 2.1.

Wykres został wykonany dla populacji USA z roku 1959. Jak można zauważyć, w środkowym przedziale wiekowym $a \in (30, 90)$ zależność ta jest zależnością wykładniczą (skala logarytmiczna). W tym też przedziale dopasowano do danych krzywą wykładniczą. Najłatwiejsze co do oceny przyczyn odstępstwo od prawa Gomperta, to drobne maksimum w okolicach $a = 20$. Wnioski na temat przyczyn śmiertelności w tym przedziale wiekowym można wyciągnąć na podstawie analizy statystyk przyczyn śmierci (dostępne między innymi w [26]). Wynika z nich iż przyczyną tą jest większa liczba wypadków w szczególności wypadków drogowych (okres uzyskiwania prawa jazdy).

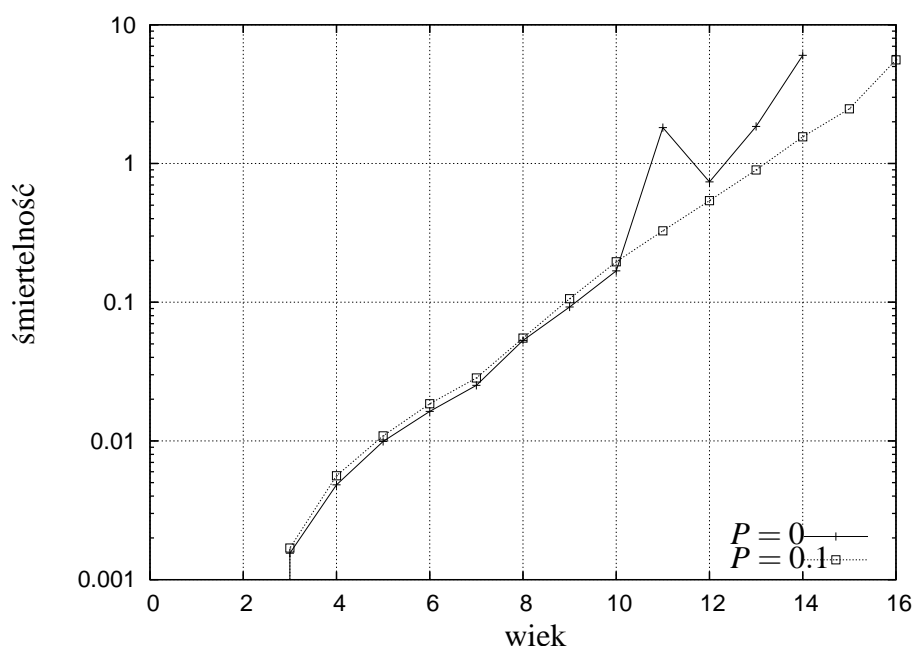
Nieco trudniejsza w ocenie jest kwestia *plateau* pojawiającego się w okolicach $a = 90$. Być może przyczyną tego faktu jest dostęp do opieki lekarskiej, który powoduje mniejszą śmiertelność w tym przedziale wiekowym. Analityczne rozwiązanie nieco zmodyfikowanego modelu Penny, na bazie którego można uzyskać analogiczne *plateau* zawiera [27].

Niełatwo jest wskazać przyczyny znacznie odbiegającej od prawa Gomperta, wysokiej śmiertelności dzieci (przedział wiekowy $0 - 10$). Przypuszczalnie w tym przedziale, prawo Gomperta nie obowiązuje z powodów nie uwzględnianych w opisywanym modelu.

Podstawową zaletą modelu Penny, w porównaniu ze zbyt uproszczonym modelem logistycznym, jest fakt iż możemy przy jego użyciu uzyskać wykładniczą zależność śmiertelności od wieku. Przykłady takiej zależności uzyskanej w wyniku symulacji zawierają rysunki 1.4 i 1.3 z poprzedniego rozdziału. Jednak aby zależność ta była odtworzona, powinny być spełnione następujące warunki:

- $M > P$ – w przeciwnym przypadku nie dostaniemy efektu kumulowania się mutacji,
- $P > 0$ – by uniknąć efektu ukazanego na rysunku 2.2,
- pod uwagę należy brać śmiertelność z przyczyn genetycznych.

Ostatni warunek nie jest konieczny, lecz powoduje iż cały obszar $q(a)$ spełnia prawo Gompertza, a nie tylko jego część (opisano dokładniej w podrozdziale 3.2). Wykres na rysunku 2.2 zawiera porównanie wyników przykładowej symulacji dla $P = 0$ i $P = 0.1$ ($M = 1$ w obu przypadkach). Dla $P = 0$ wykładniczą zależność śmiertelności od wieku zaburza ostre maksimum. Przyczynami o opisie tego zjawiska zajmiemy się w dalszej części pracy.



Rysunek 2.2: Wyniki przykładowej symulacji dla $P = 0$ i $P = 0.1$ – demonstracja niepożądanego efektu.

2.2.2 Zestawienie wyników symulacji z danymi demograficznymi

W celu zweryfikowania naszej implementacji algorytmu modelu Penny, odtworzymy teraz powszechnie znane wyniki, wykorzystując w tym celu program opisany w rozdziale 3.1 (jego kluczowe fragmenty zawiera dodatek B).

Zanim dokonamy próby zestawienia wyników symulacji i danych doświadczalnych, zdefiniujemy śmiertelność całkowitą $q(a)$. Ponieważ nie zawsze mamy pewność w rozpatrywanych przypadkach – zarówno w przypadku wyników symulacji jak i danych demograficznych – czy obserwujemy stan stacjonarny (a

w przypadku rzeczywistej populacji mamy wręcz pewność że nie jest to stan stacjonarny), dotychczasowa definicja $q(a)$:

$$q(a) = 1 - s(a) = \frac{n(a) - n(a+1)}{n(a)}, \quad (2.2)$$

okazuje się niewystarczająca. W niektórych przypadkach, w stanie niestacjonarnym możemy otrzymać $q(a) < 0$, gdy założenie $n(a, t) \approx n(a, t+1)$ okaże się niespełnione. Dlatego też by wyeliminować tę niedogodność, określimy $q(a)$ jako:

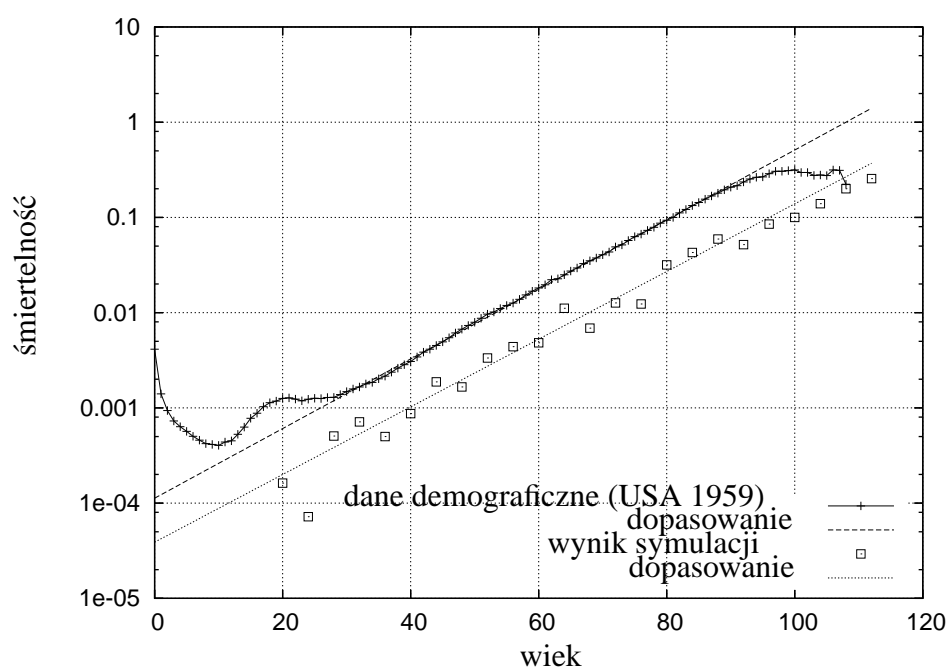
$$q(a) = \frac{n(a, t) - n(a+1, t+1)}{n(a)}. \quad (2.3)$$

Jako dane porównawcze posłużą nam dane demograficzne dla USA z roku 1959. Aby porównać śmiertelność otrzymaną w wyniku symulacji z przypadkiem rzeczywistym, dobierzemy najpierw odpowiednio parametry symulacji. Ponieważ rozrodczość brana pod uwagę powinna być raczej średnią z wielu lat niż wartością chwilową, po uwzględnieniu danych demograficznych dobraliśmy parametr $B = 0.08$. Jako jednostkę czasu przyjęliśmy 1 rok. Limit wieku w naszym przypadku wynosi 32 (długość genomu $L = 32$ bity) i w praktyce nie jest on osiągany. Dlatego też musieliśmy przeskalować wiek symulowany tak, by odpowiadał rzeczywistym danym. Po przemnożeniu wieku a o czynnik $v = 4$, otrzymamy górną, nieosiągalną granicę wieku jako 128 co dość dobrze odpowiada rzeczywistości. Jako minimalny wiek rozrodczy przyjęliśmy granicę 16 lat (co odpowiada parametrowi $R = 4$). Ponieważ parametrów takich jak M, P , i T dla populacji rzeczywistej nie znamy, dobraliśmy je empirycznie (spośród zestawu różnych wartości tych parametrów wybraliśmy taki, który daje wynik najbardziej zbliżony do danych doświadczalnych, w sensie odległości średniokwadratowej). Pojemność środowiska nie odgrywa większej roli dla wyników końcowych, dlatego przyjęliśmy ją tak, by uzyskać rozsądną statystykę ($N = 10^6$).

Porównanie zależności śmiertelności od wieku – wyników przeprowadzonej symulacji i danych demograficznych – przedstawia wykres na rysunku 2.3.

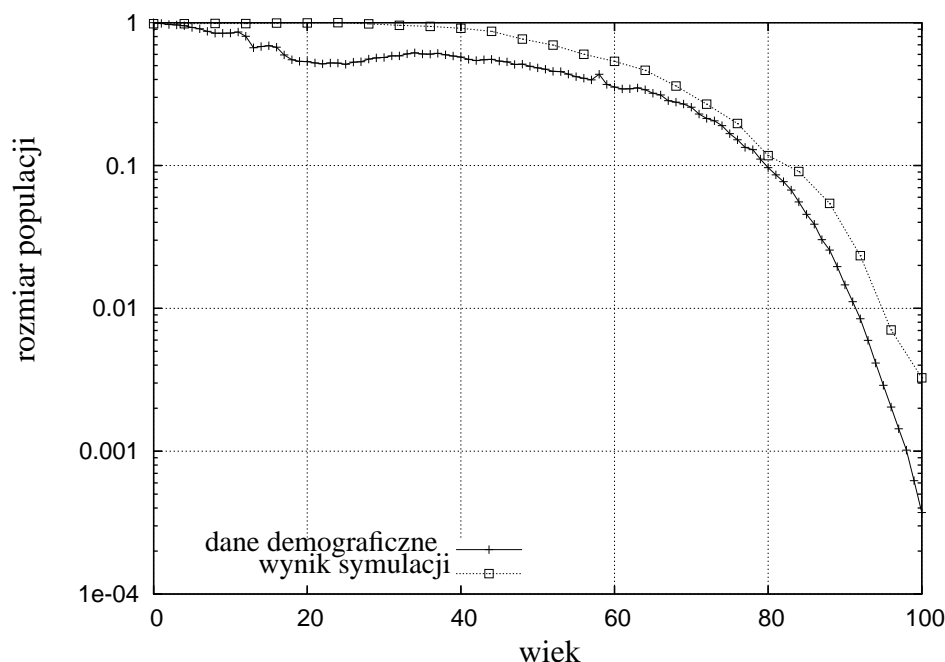
Jak można zauważyć nachylenie obu prostych na wykresie logarytmicznym jest zbliżone. Śmiertelność jest nieco niższa w przypadku wyników symulacji, lecz należy pamiętać iż bierzemy pod uwagę w niej jedynie genetyczne przyczyny śmierci, zaś dane demograficzne dotyczą śmiertelności całkowitej. Rozbieżności obserwujemy zarówno dla osobników najmłodszych (zerowa śmiertelność w przypadku symulacji), jak i osobników najstarszych (brak *plateau*). Nie mniej jednak w obszarze w którym obowiązuje prawo Gomperta, symulacja odtwarza rzeczywistość.

Odtworzenie zależności $n(a)$ przysparza dużo więcej trudności. Jest ona znacznie bardziej podatna na wszelkie chwilowe wahania i ich efekty są widoczne nawet bardzo długo po ustąpieniu bodźców zakłócających, czego przykładem są



Rysunek 2.3: Porównanie wyników symulacji (standardowy model Penny $(N, B, R, M, P, T) = (10^6, 0.08, 4, 0.1, 0.02, 3)$) z danymi demograficznymi (USA rok 1959, źródło [1]).

współczesne populacje europejskie, w których strukturze wiekowej obecne są ślady II wojny światowej. Zależy więc od wielu czynników zewnętrznych, których w naszym modelu nie jesteśmy w stanie uwzględnić. Aby mimo tego pokazać, że ogólny charakter tej zależności jest zbliżony do rzeczywistej, umieściliśmy wynik symulacji razem z danymi demograficznymi na wykresie 2.4. Przedstawia on zależność znormalizowanej liczby osobników od wieku ($n(a)/n_{max}$).



Rysunek 2.4: Porównanie zależności rozmiaru populacji od wieku dla danych demograficznych (USA 1959, źródło [1]), oraz wyników symulacji (standardowy model Penny $(N, B, R, M, P, T) = (10^6, 0.08, 4, 0.3, 0.02, 3)$)

Pomimo, że efekt nie jest w pełni zadowalający, to ogólny charakter krzywej $n(a)$ jest zachowany. Uwagę zwraca “załamanie” krzywej dla osobników powyżej wieku $a = 60$, odbiegające od krzywej wykładniczej, którego nie odtwarza model logistyczny.

Rozdział 3

Symulacje komputerowe

3.1 Podstawowy algorytm symulacji komputerowej

3.1.1 Opis implementacji algorytmu

Wszystkie prezentowane w pracy wyniki symulacji, zostały uzyskane przy użyciu programu napisanego przez autora. Algorytm modelu Penny został zaimplementowany w języku Java [28] (jego najbardziej istotne fragmenty zostały umieszczone w dodatku B). Przeciętnie, czas pojedynczej symulacji (na komputerze klasy Pentium III) wynosi ok 4 godzin (5000 kroków czasowych, $N \approx 10^6$).

Wybór języka programowania, podyktowany był kilkoma względami. Pierwszy z nich to elastyczność samego języka, która pozwala na łatwe modyfikacje istniejących programów, oraz szybką implementację omawianego algorytmu. Nie bez znaczenia pozostają też inne cechy Javy, takie jak wygodne środowisko wykonawcze, automatyczne zarządzanie pamięcią, kontrola zakresów tablic itp. Związany z tym narzut na wymagany czas procesora, można zminimalizować dzięki mechanizmom takim jak *Just in Time Compilation* [29], [30], czy nawet statycznej kompilacji do kodu natywnego, na przykład przy użyciu kompilatora GCJ [31].

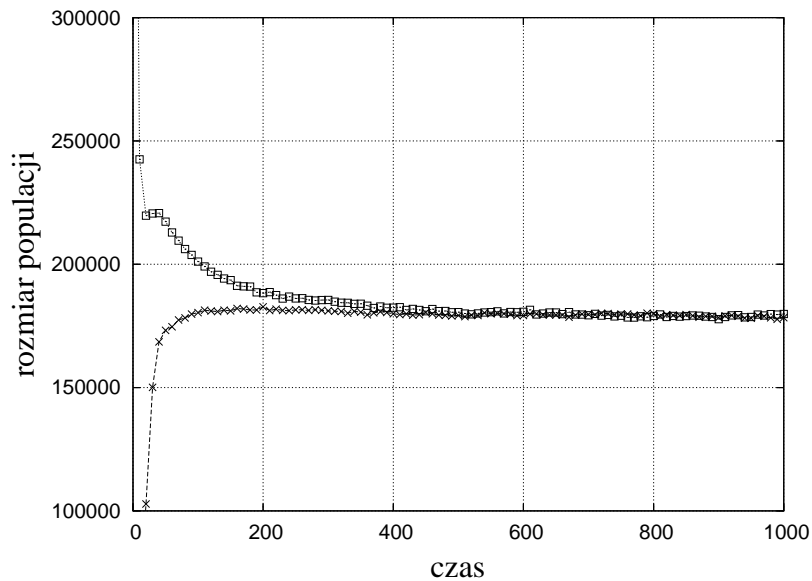
Program do przechowywania obiektów reprezentujących poszczególne osobniki, używa listy dwukierunkowej. Genom pojedynczego osobnika reprezentowany jest przez zbiór bitów, realizowany przez klasę biblioteczną Javy. Program pozwala na określenie warunków początkowych symulacji (rozkład mutacji w genomach), oraz zapis i odczyt bieżącego stanu symulacji.

Uzyskiwane wyniki symulacji są powtarzalne – podczas uruchamiania programu dla tego samego zestawu parametrów lecz z inną wartością inicjującą generator liczb pseudolosowych, główne parametry charakteryzujące populację są do siebie zbliżone (w sensie wartości średnich). Jakkolwiek, zmianom ulega położenie charakterystycznych maksimów w funkcji rozkładu mutacji (opisanych w

podrozdziale 3.3).

3.1.2 Ocena warunków symulacji

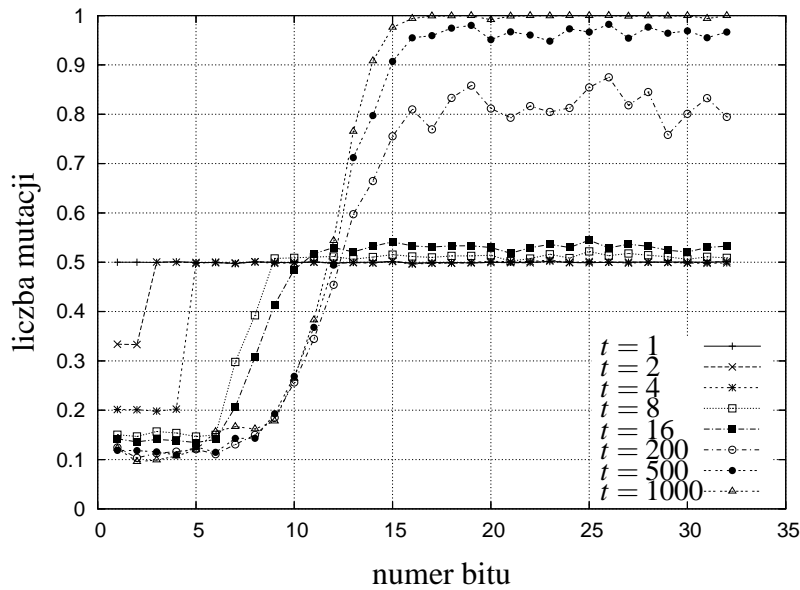
Dla analizy końcowych wyników symulacji istotne jest to, jaki wpływ mają na nie warunki początkowe, czyli rozkład mutacji w genomach osobników stanowiących populację startową. Ponieważ określenie jakie warunki początkowe panowały w rzeczywistości nie wydaje się możliwe do realizacji (już samo wyznaczenie “punktu początkowego” nastrocza wiele trudności), oczekujemy od modelu, iż jego podstawowe parametry nie będą zależeć od wspomnianych warunków. Do prostego testu, posłużą nam wyniki symulacji dla przypadku z genomami początkowymi bez mutacji, oraz z genomami w których każdy bit jest z prawdopodobieństwem $P_m = 0.5$ ustawiony na '1'.



Rysunek 3.1: Porównanie $n(t)$ dla różnych warunków początkowych: '□' – wyniki dla symulacji z genomami startowymi bez mutacji, '×' – wyniki dla genomów z (średnio) połową bitów ustawioną na starcie na '1'.

Wykres na rysunku 3.1 przedstawia zależność rozmiaru populacji od czasu $n(t)$ dla owych dwóch przypadków. Symulacje zostały przeprowadzone dla identycznych zestawów parametrów (lecz z różnymi genomami startowymi). Dla chwili czasowej $t = 0$ (punkt niewidoczny na wykresie) rozmiary populacji były takie same ($n(0) = 6 \cdot 10^5$). W wersji z genomami “czystymi” następuje powolny spadek populacji aż do osiągnięcia n_{st} w stanie stacjonarnym. W drugim przypad-

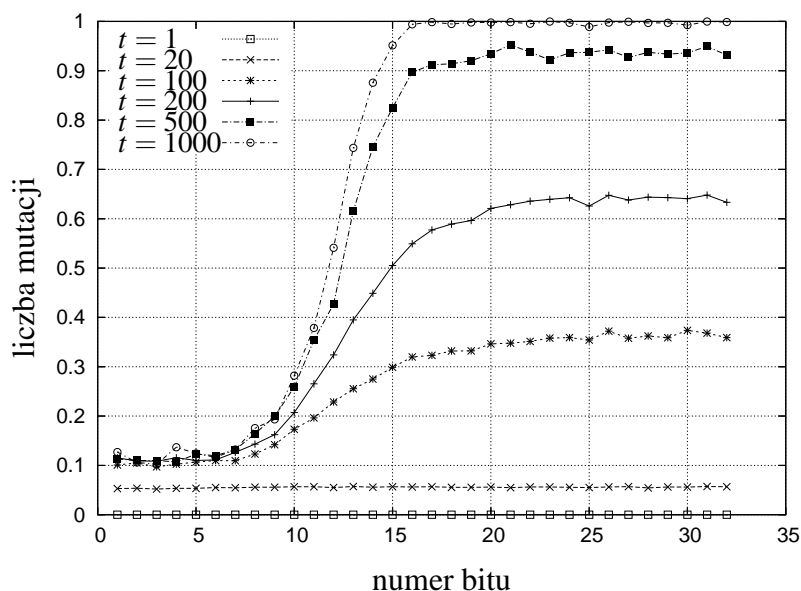
ku, po gwałtownym spadku w pierwszych krokach populacja powoli wzrasta, by następnie dążyć do takiego samego rozmiaru n_{st} . Jest to typowe zachowanie układu dla różnych rozkładów początkowych mutacji. Sytuacje w których zachowują się one odmiennie, występują gdy startowe genomy w którejś z symulacji mają nadmierną liczbę mutacji w wyniku czego populacja wymiera¹. Aby dokładniej poznać jak przebiega proces dochodzenia do stanu stacjonarnego w obu przypadkach, przeanalizujemy sposób w jaki zmienia się w czasie funkcja $m(k)$ dla obu zaprezentowanych przypadków.



Rysunek 3.2: Funkcja $m(k)$ dla różnych kroków czasowych. Populacja startowa ($t = 0$) posiada genomy z prawdopodobieństwem wystąpienia mutacji $P_m = 0.5$ dla każdego bitu. Standardowy model Penny $(N, B, R, M, P, T) = (10^6, 0.8, 6, 1, 0, 2)$

Wykresy na rysunkach 3.2 i 3.3 przedstawiają $m(k)$ dla różnych etapów symulacji. Dla kroku czasowego $t = 1$, $m(k) = \text{const.}$ i jest równe odpowiednio ≈ 0.5 oraz 0. Rozpatrzmy przypadek pierwszy ($P_m = 0.5$). Ponieważ w prezentowanej symulacji próg mutacji aktywnych wynosi $T = 2$, w pierwszym kroku czasowym gdy odsłaniana test tylko pierwsza pozycja w genomie, $m(k)$ pozostaje bez zmian. Eliminacja następuje tylko z powodu czynnika Verhulsta, a więc niezależnie od genomu osobnika. Dla wyników w kroku $t = 2$ istotne będą dwie pierwsze pozycje w genomie.

¹w innych przypadkach rozbieżności nie udało się zaobserwować.



Rysunek 3.3: Funkcja $m(k)$ dla różnych kroków czasowych. Populacja startowa ($t = 0$) posiada genomy bez mutacji. Standardowy model Penny (N, B, R, M, P, T) = $(10^6, 0.8, 6, 1, 0, 2)$

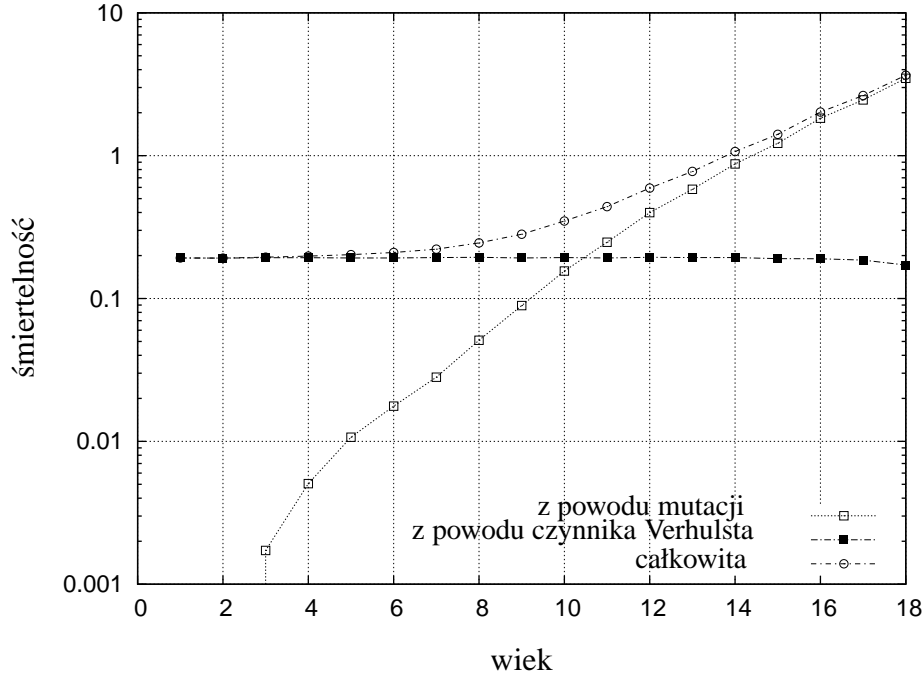
	$t = 1$	$t = 2$
Dwa pierwsze bity genomu	Prawdopodobieństwo wystąpienia	
(0,0)	1/4	1/3
(0,1)	1/4	1/3
(1,0)	1/4	1/3
(1,1)	1/4	0

Tabela 3.1: Prawdopodobieństwo wystąpienia poszczególnych konfiguracji dwóch pierwszych bitów genomu (dla kroków czasowych $t = 1$ i $t = 2$), dla genomów populacji prezentowanej na wykresie 3.2

W obecnej sytuacji mamy do czynienia z czterema przypadkami: $(0, 0)$, $(0, 1)$, $(1, 0)$, $(1, 1)$, z których każdy występuje w kroku czasowym $t = 1$ z takim samym prawdopodobieństwem równym $\frac{1}{4}$, co przedstawiono w tabeli 3.1. W trakcie drugiego kroku zostaną wyeliminowane wszystkie osobniki z początkowymi pozycjami $(1, 1)$. Tak więc na pierwszej i drugiej pozycji dostaniemy '1' z prawdopodobieństwem równym $\frac{1}{3}$ co widać również na prezentowanym wykresie (rys. 3.2). Analogiczna sytuacja wystąpi również dla kroków następnych – na przykład w kroku $t = 4$ z początkowych 16 wariantów na czterech pierwszych pozycjach, zostaje pięć (cztery z pojedynczymi mutacjami i jeden bez mutacji). Tę prostą statystykę zaburza nam w kroku $t = 6$ pojawienie się nowych osobników (minimalny wiek reprodukcji $R = 6$). W drugim z prezentowanych przypadków ($P_m = 0$, rysunek 3.3) funkcja $m(k)$ osiąga dla stanu stacjonarnego te same wartości jak poprzednio, pomimo braku mutacji dla $t = 0$ – mutacje pojawiają się tylko podczas przekazywania genomu potomkom.

Następną istotną kwestią dla analizy wyników jest to, jak długość genomu L pojedynczego osobnika, wpływa na wynik końcowy. Wartość ta determinuje maksymalny wiek w naszej symulacji ($a_{max} = L$ choć w praktyce, w stanie stacjonarnym jest on na ogół o około połowę mniejszy), co dla wartości $L = 32$ powoduje, iż otrzymane rezultaty (jak np. $n(a)$) musimy skalować jeśli chcemy je porównać z danymi demograficznymi. Analizę wpływu L na zachowanie się badanego układu przedstawiono w publikacji [32]. Mimo iż trudno w tej kwestii o jednoznaczne wnioski, wydaje się iż zasadnicze cechy modelu (jak choćby wykładniczy wzrost śmiertelności) nie ulegną zmianie wraz z $L \rightarrow \infty$. Wszystkie przedstawione w niniejszej pracy symulacje, zostały przeprowadzone dla $L = 32$.

Również ważnym zadaniem jest ocena tego czy układ osiągnął stan stacjonarny. Na ogół wystarczającym, jest kryterium opisane równaniem (1.5) w podrozdziale 1.4.1. Jakkolwiek dla zaobserwowania efektów opisanych w podrozdziale 3.3, to kryterium okazuje się niewystarczające. Osobną kwestią jest również to, jak symulowany układ w stanie stacjonarnym odtwarza rzeczywistość, tzn. czy możemy stwierdzić jednoznacznie, że obecna populacja (ludzka) znajduje się w stanie stacjonarnym. Ponieważ ze względu na ewolucyjny charakter jej rozwoju należy przypuszczać iż jest ona obecnie daleka od stanu stacjonarnego (na co wskazuje również zmieniająca się w czasie struktura wiekowa populacji), co podważa zasadność porównania wyników symulacji w stanie stacjonarnym z danymi demograficznymi (zaś użycie wyników nie pochodzących ze stanu stacjonarnego uzależnia nas od populacji początkowej). Nie mniej jednak, odpowiednio dobierając parametry modelu, możemy taką próbę porównania podjąć. Problem ten został nieco dokładniej poruszony w artykule [22].



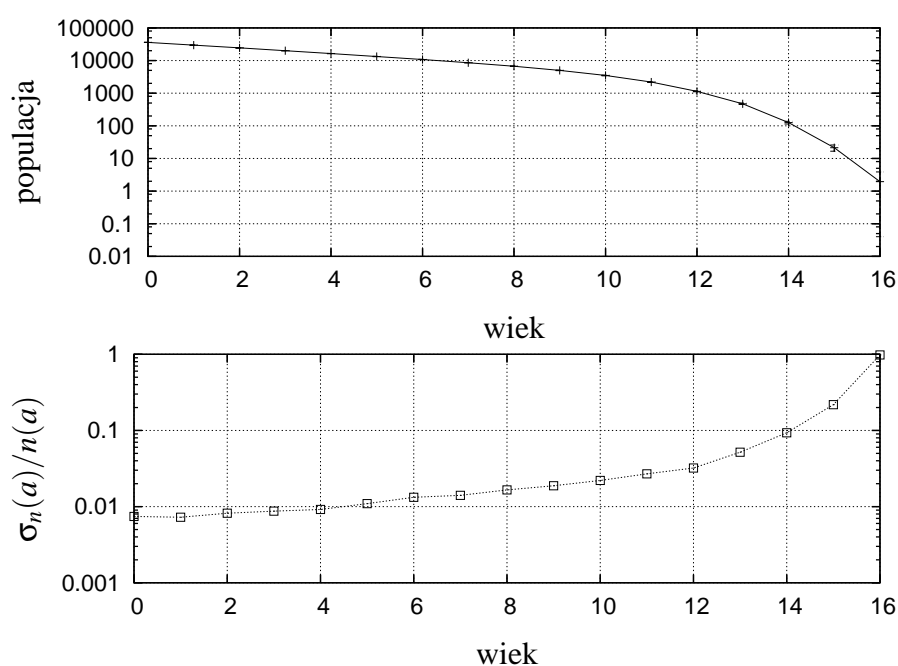
Rysunek 3.4: Zależność śmiertelności od wieku dla różnych sposobów zliczania (rezultat przykładowej symulacji).

3.1.3 Oszacowanie błędu wyników

Jak zostało wspomniane we wcześniejszej części pracy, prezentowane wyniki zostały uzyskane poprzez uśrednienie badanych wielkości po całej populacji, po czym z obliczonych średnich, została wyliczona średnia po 20 krokach czasowych. Powodem takiego wyboru jest z jednej strony chęć uchwycenia wartości chwilowych, z drugiej zaś uniknięcia wpływu fluktuacji na wynik końcowy. Przyjmując standardowe dla tego typu metod oszacowanie $\text{err} \sim 1/\sqrt{N}$, dostajemy rozsądną wartość błędu. Kiedy natomiast przyjrzymy się jak wygląda wyliczone odchylenie standardowe, zauważymy że jego wartość jest w pełni akceptowalna. Przykład typowych wyników uzyskanych na bazie modelu Penny, przedstawiają wykresy na rysunku 3.5. Jak można zaobserwować na dolnym wykresie, wielkość:

$$\frac{\sigma_n(a)}{n(a)}, \quad (3.1)$$

rośnie monotonicznie wraz z wiekiem. Duże wartości osiąga jedynie dla wysokich a , co jest zrozumiałe jeśli weźmiemy pod uwagę fakt, że w górnym przedziale wiekowym, liczba osobników jest znikoma, a przez to dysponujemy ubogą statystyką.



Rysunek 3.5: Typowe wyniki symulacji dla modelu Penny. Górny wykres przedstawia zależność $n(a)$ (uśrednioną), wraz ze słupkami błędów (odchylenie standardowe), które widoczne są jedynie dla wartości a z górnej części prezentowanego przedziału. Na dolnym wykresie znajduje się zależność stosunku odchylenia standardowego wielkości n do niej samej, w zależności od a

3.2 Analiza słuszności prawa Gompertza dla różnych mechanizmów śmierci

Jak wspomniano w poprzedniej części pracy, model Penny pozwala na uzyskanie jako wyniku symulacji, populacji dla której zależność śmiertelności od wieku jest zgodna z empirycznym prawem Gompertza [12]. Tę zgodność osiągamy jeśli uwzględnimy tylko przypadki śmierci z przyczyn genetycznych (przekroczenie progu T przez liczbę aktywnych mutacji). Aby wyznaczyć śmiertelność z przyczyn genetycznych q_m , wyznaczymy $n(t+1, a+1)$:

$$n(t+1, a+1) = n(a) - k_v(a) - k_m(a), \quad (3.2)$$

gdzie $k_v(a)$ oznacza liczbę przypadków śmierci z powodu czynnika Verhulsta, zaś $k_m(a)$ – z przyczyn genetycznych. Ponieważ rozważamy populację w stanie stacjonarnym, możemy wyznaczyć współczynnik przeżycia s jako:

$$s(a) = \frac{n(a+1)}{n(a)} = 1 - \frac{k_v(a) + k_m(a)}{n(a)}. \quad (3.3)$$

Stąd:

$$q(a) = \frac{k_v(a) + k_m(a)}{n(a)}, \quad (3.4)$$

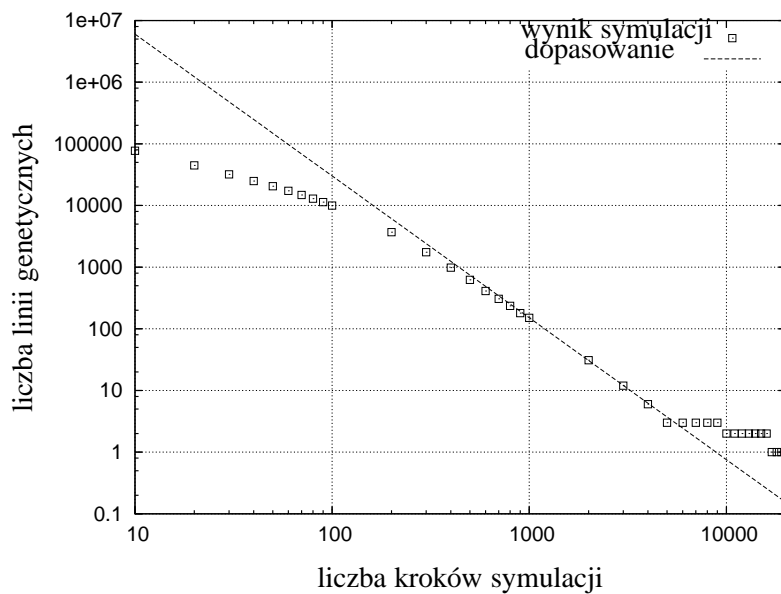
i ostatecznie śmiertelność powodowana mutacjami ($k_v = 0$):

$$q_m(a) = \frac{k_m(a)}{n(a)} \quad (3.5)$$

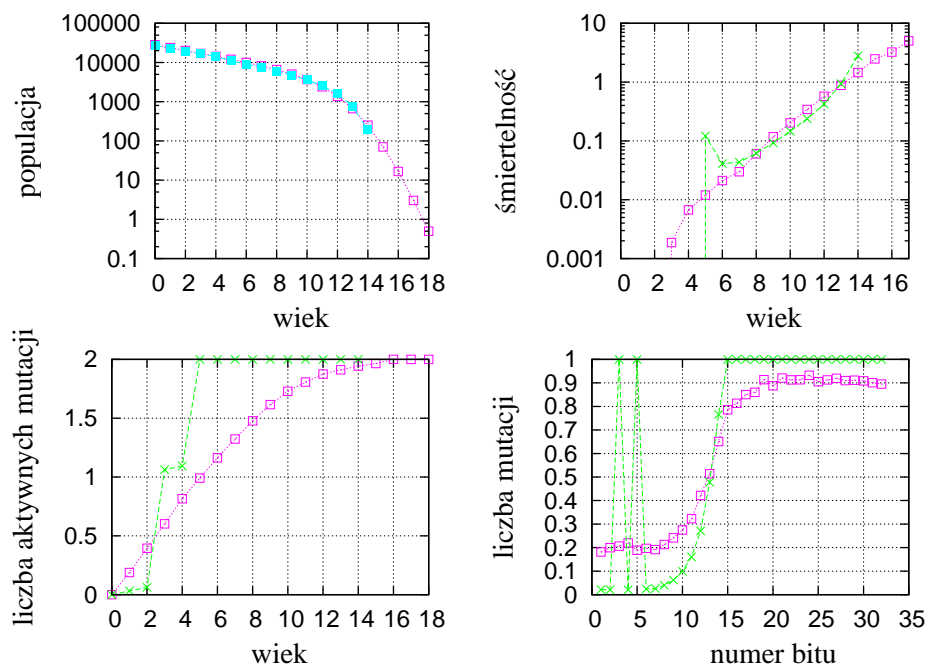
Wykres na rysunku 3.4 przedstawia zależności $q(a)$, $q_v(a)$ i $q_m(a)$ otrzymane w wyniku symulacji. Zależność $q_v(a)$ pozostaje (w przybliżeniu) stała, zachowując się jak śmiertelność dla modelu logistycznego. Dodanie do $q_m(a)$ części pochodzącej od czynnika Verhulsta, powoduje iż o charakterze wykładniczym $q(a)$ możemy mówić jedynie dla dużych a .

3.3 Dynamika redukcji rodzin w modelu seksualnym i aseksualnym

Przedstawimy teraz analizę problemu spadku liczby linii genetycznych, wraz z wynikami symulacji. W tym celu, przebadamy zachowanie się liczby linii genetycznych w funkcji czasu – $G(t)$, zarówno dla modelu aseksualnego jak i dla seksualnego, a następnie wprowadzimy dodatkową wielkość charakteryzującą różnorodność genomów w populacji – funkcję rozrzutu.



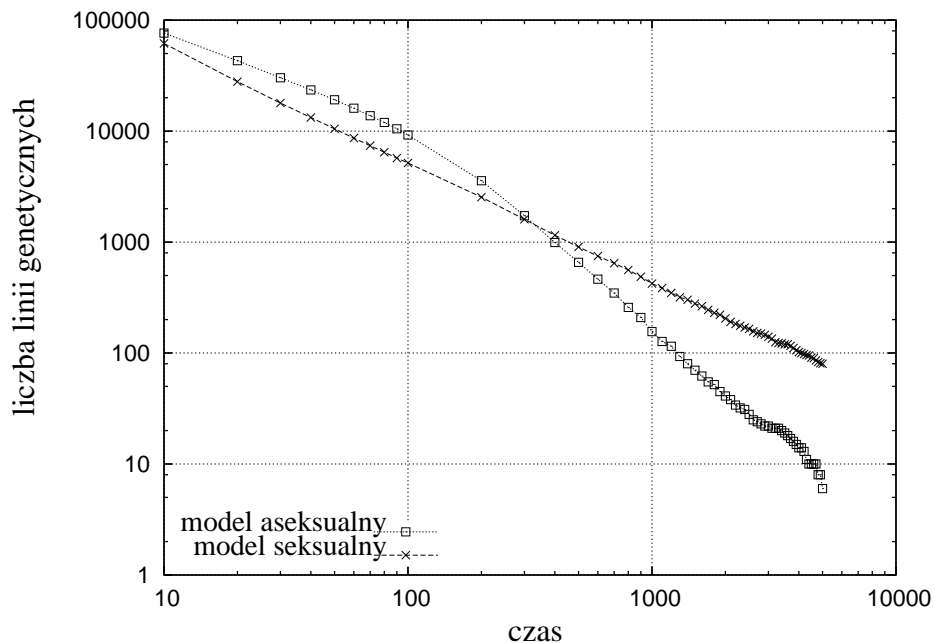
Rysunek 3.6: Liczba linii genetycznych w funkcji liczby kroków symulacji. Standardowy model Penny $(N, B, R, M, P, T) = (10^6, 1, 7, 1, 0, 3)$



Rysunek 3.7: Struktura populacji – wynik symulacji $(N, B, R, M, P, T) = (10^6, 1.0, 7, 1, 0, 3)$, uzyskany po 500 (\square) i 20000 (\times) krokach czasowych.

Wykres na rysunku 3.6 przedstawia zależność liczby linii genetycznych w populacji od kroku czasowego $G(t)$ (podwójna skala logarytmiczna), otrzymaną w wyniku symulacji dla typowego zestawu parametrów standardowego (aseksualnego) modelu Penny. Jak widać na przedstawionym wykresie, zależność ta ma charakter malejący. Można ją podzielić na trzy regiony. W pierwszym regionie, obejmującym czas od początku symulacji do około kilkuset kroków czasowych, ma ona charakter funkcji potęgowej $G(t) = a \cdot t^u$, gdzie a i u są stałymi, przy czym $u \approx -1$. Następnie funkcja ta gładko przechodzi w drugi region, gdzie jej zależność od czasu ma ten sam charakter, lecz stała $u \approx -2$ (w tym rejonie zależność ta pozostaje najdłużej). Stan ten utrzymuje się przez na ogół kilka tysięcy kroków, aż osiąga rejon trzeci – przechodzenia w $G(t) = 1$.

Charakterystyczne dla modelu jest to, że liczba linii genetycznych zmierza we wszystkich zaobserwowanych przypadkach do 1 o ile czas symulacji jest dostatecznie długi. Należy przypuszczać, iż dużą rolę w tym zjawisku odgrywa czynnik Verhulsta, w naszym modelu jest to jedyny możliwy sposób w jaki osobniki mogą wpływać na siebie wzajemnie. Gdy podczas symulacji jedna z linii genetycznych w wyniku fluktuacji uzyska przypadkową niewielką przewagę nad innymi, może łatwiej wypełniać środowisko nowymi osobnikami – śmierć spowodowana przepełnieniem środowiska nie zależy od genomu osobnika – nie czyni więc tak dużych zniszczeń jak w przypadku mniej licznych podpopulacji.



Rysunek 3.8: Porównanie ewolucji $G(t)$ dla modelu seksualnego i aseksualnego.

Odmienne $G(t)$ zachowuje się dla modelu seksualnego. Porównanie wyników dla obydwu modeli zawiera wykres na rysunku 3.8. O ile w przypadku standardowego, aseksualnego modelu Penny, funkcja ta potrzebuje kilkuset kroków po których zmienia swój charakter na opisany powyżej, to dla modelu uwzględniającego płeć osobnika zachowuje się ona jednolicie w całym przedziale t . Co więcej, w odróżnieniu od modelu aseksualnego charakter krzywej co prawda pozostaje ten sam: $G(t) = a \cdot t^u$, lecz stała u jest tu inna: $u \approx -1$. Różnica ta, nie była wcześniej dostrzegana w literaturze, gdzie uznawany był dotąd wynik $u \approx -1$ dla obydwu modeli (jak choćby w [13]). Powodem takiego wyniku, była zbyt mała pojemność środowiska, a co za tym idzie i rozmiar populacji w trakcie symulacji. Gdy N nie jest dostatecznie duże, liczba linii genetycznych osiąga wartość 1 zanim $G(t)$ ma szansę “dotrzeć” do regionu w którym u osiąga wartość w okolicach -2 . Problem ten został przez nas przedyskutowany w publikacji [33].

Drugą wielkością pozwalającą wnioskować o tym jak różnorodne są genomy w populacji, jest funkcja rozkładu mutacji $m(k)$. Dolny, prawy wykres na rysunku 3.7 przedstawia $m(k)$ – wynik symulacji uzyskany po 500 oraz 20000 krokach czasowych (wynik reprezentuje rezultat tej samej symulacji dla której wykonany wykres na rys. 3.6). Charakterystyczne dla wyników po wielu krokach czasowych oraz $P \approx 0$ są ostre maksima pojawiające się w początkowym obszarze wspomnianej funkcji – dla prezentowanego przypadku osiąga ona wartości $m(k) = 1$ dla bitów na pozycjach $k = 3$ i $k = 5$. Pozycje na których one występują, nie są jednoznacznie związane z parametrami symulacji, dla tych samych parametrów dostajemy różne pozycje maksimów w zależności od wartości startowej generatora liczb pseudolosowych. W omawianym przypadku osiągają one wartości równe 1, co oznacza że na owych dwóch pozycjach, mutacje posiadają wszystkie osobniki w populacji. Ponieważ dla prezentowanej symulacji parametr $T = 3$, a minimalny wiek reprodukcji R wynosi 7, wartości $m(k)$ na wszystkich pozostałych bitach od pierwszego do siódmego mają wartości zbliżone do zera (efekt selekcji ewolucyjnej – osobniki muszą przeżyć i osiągnąć wiek rozrodczy). Niewielki “szum” powodujący iż nie są to dokładnie zera, pochodzi od osobników które nie przeżyją swojego genomu dalej (podczas narodzin uzyskały one dodatkową mutację w tym newralgicznym obszarze). Jak można zauważyć utrzymuje się on w tym obszarze na stałym poziomie dla wszystkich bitów.

Z przedstawionych wykresów wynika, iż dla odpowiednio długiego czasu symulacji genomy populacji stają się w przeważającej części jednorodne. Za wyjątkiem kilku pozycji (w naszym przypadku pozycje $k \in (9, 14)$), przyjmują z dużym bardzo prawdopodobieństwem bądź zera (początkowy obszar genomu z wyjątkiem maksimów), bądź też jedynki (maksima oraz pozycje od 15 bitu). Sytuacja ta wpływa również na pozostałe wielkości charakteryzujące populację. Rozkład mutacji aktywnych $\mu(a)$ zmienia się w sposób skokowy (lewy dolny wykres na rysunku 3.7). Fakt iż występują wspomniane maksima w funkcji $m(k)$, znajduje

również odzwierciedlenie w zależności od śmiertelności z przyczyn genetycznych od wieku (prawy górny wykres). Po 20000 kroków $q_m(a)$ również posiada maksimum dla $a = 5$, w przeciwieństwie do tej samej zależności uzyskanej po 500 krokach symulacji. Charakterystyczne jest również skrócenie maksymalnego wieku osobników występujących w populacji (widoczne między innymi w zależności $n(a)$ – lewy górny wykres).

Omówione powyżej wyniki wskazują na to, iż wprowadzone w podrozdziale 1.4.1 rozróżnienie na stan stacjonarny i niestacjonarny (wzór 1.5) okazuje się niewystarczające, gdy rozpatrujemy takie wielkości jak omawiana $G(t)$. Mimo iż układ na ogół już po kilkuset krokach osiąga stan w którym całkowita liczba osobników w populacji nie zmienia się w czasie (prezentowany wcześniej wykres na rysunku 3.1), ta liczba kroków okazuje się niewystarczająca dla zależności takich jak $m(k)$ czy $q_m(a)$ – dla różnych wielkości obowiązują różne skale czasowe. Stąd też bardziej odpowiednim wyznacznikiem tego czy nasz badany układ osiągnął stan stacjonarny wydaje się być liczba linii genetycznych. Ponieważ po osiągnięciu jednej linii genetycznej w populacji, nie ulegnie już ona zmianie, stan taki uznajemy za stacjonarny.

Dokładniejszym kryterium było by wprowadzenie żądania, aby badać estymatory wartości oczekiwanych wszystkich omawianych tu wielkości charakteryzujących populację, lecz nie dość że komplikuje to znacznie określanie czy stan obserwowany jest stanem stacjonarnym, to i tak nie daje nam całkowitej pewności, że jakaś wielkość której dotąd nie uwzględniliśmy nie zmieni naszego postrzegania stacjonarności układu. Z drugiej strony, znacznie ciekawsze pod względem badawczym wydaje się zmierzanie układu do stanu stacjonarnego, niż sam układ w stanie stacjonarnym.

W celu zbadania jak skorelowana jest liczba linii genetycznych z różnorodnością genomów w populacji, wprowadzimy wielkość charakteryzującą tę drugą cechę – funkcję rozrzutu $p(t)$. Jako miarę tej różnorodności dla pozycji k w genomie, przyjmijmy wariancję z wartości bitu na tej pozycji liczoną po wszystkich osobnikach, czyli:

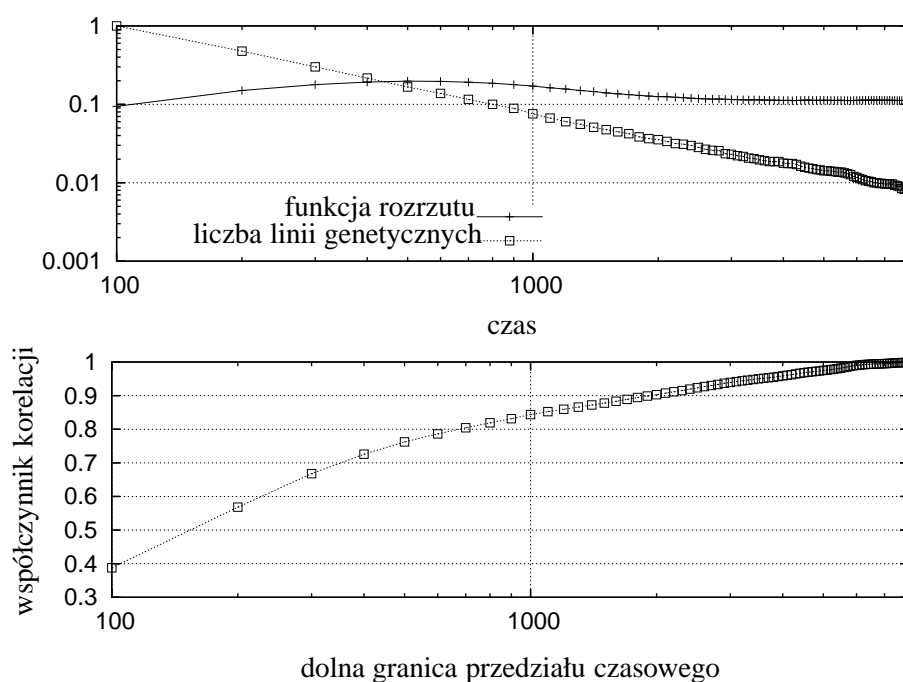
$$\sigma^2(k) = \frac{1}{n} \sum_{i=1}^n \left(g(i, k) - \overline{g(k)} \right)^2, \quad (3.6)$$

gdzie $g(i, k)$ oznacza wartość bitu genomu i -tego osobnika na pozycji k , n jest liczbą osobników w bieżącym kroku czasowym, zaś

$$\overline{g(k)} = \frac{1}{n} \sum_{i=1}^n g(i, k), \quad (3.7)$$

jest średnią arytmetyczną.² Jeśli otrzymaną w określonym kroku czasowym t wa-

²W tym i w poniższych wzorach, kreska nad funkcją zawsze oznacza średnią liczoną po wszystkich osobnikach w populacji, o ile nie zaznaczono że jest inaczej.



Rysunek 3.9: Wykres górny: liczba linii genetycznych (znormalizowana) w funkcji czasu, oraz funkcja rozrzutu będąca miarą średniej odległości genomów wszystkich osobników w kroku czasowym t ; wykres dolny: współczynnik korelacji pomiędzy funkcją rozrzutu a liczbą linii genetycznych w zależności od dolnej granicy przedziału czasowego (górna granica jest stała).

riancję uśrednimy po wszystkich pozycjach k , otrzymamy funkcję rozrzutu:

$$\rho(t) = \overline{\sigma_t^2(k)} = \frac{1}{32} \sum_{k=1}^{32} \left[\frac{1}{n} \sum_{i=1}^n \left(g_t(i, k) - \overline{g_t(k)} \right)^2 \right], \quad (3.8)$$

gdzie $g_t(i, k)$ oznacza $g(i, k)$ w kroku czasowym t . Jeśli zauważymy, że $g(i, k) \in \{0, 1\}$, wówczas $\overline{(g_k^2)} = \overline{g_k}$, stąd też:

$$\sigma^2(k) = \overline{(g_k^2)} - (\overline{g_k})^2 = \overline{g_k} - (\overline{g_k})^2, \quad (3.9)$$

Zatem funkcję rozrzutu, możemy wyznaczyć na podstawie funkcji rozkładu mutacji $m(k)$:

$$\rho(t) = \frac{1}{32} \sum_{k=1}^{32} (m(k) - m^2(k)). \quad (3.10)$$

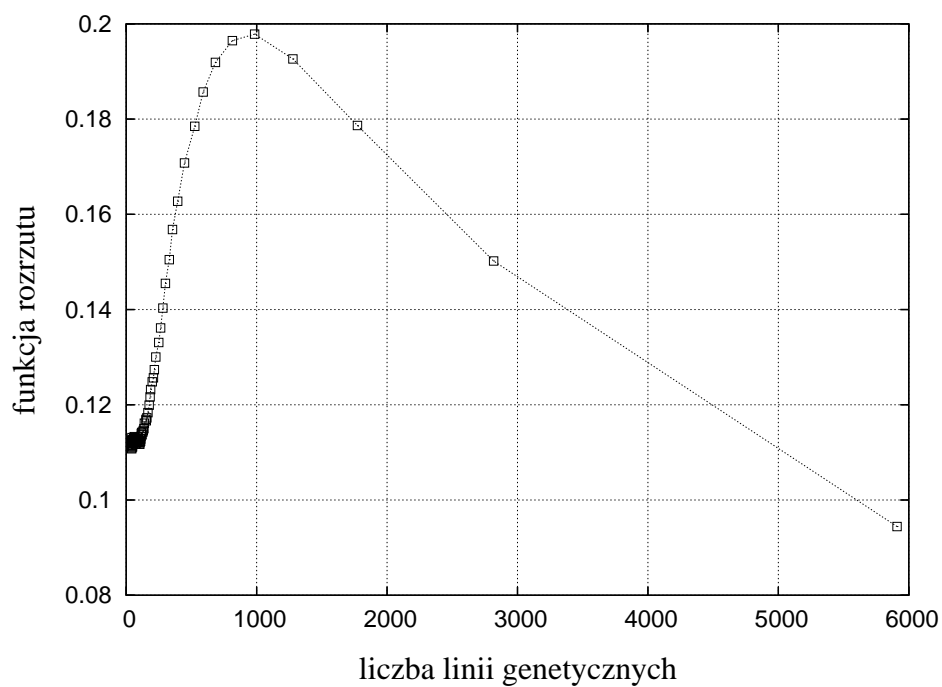
Ponieważ współczynnik korelacji pomiędzy funkcją $\rho(t)$ a liczbą linii genetycznych $G(t)$ zależy od tego jaki przedział czasowy (t_0, t_1) bierzemy pod uwagę, aby ocenić jak zmienia się stopień skorelowania tych dwóch wielkości w czasie, wyznaczmy współczynnik korelacji w zależności od lewego końca przedziału t_0 . Prawy koniec przedziału pozostaje stały $t_1 = 20000$. Zatem nasza zależność współczynnika korelacji od t_0 wygląda następująco:

$$cr(t_0) = \frac{(\overline{\rho(t_0)G(t_0)}) - \overline{\rho(t_0)} \cdot \overline{G(t_0)}}{\sqrt{\sigma_\rho^2(t_0) \sigma_G^2(t_0)}}, \quad (3.11)$$

gdzie wszystkie średnie liczone są po czasie, a σ^2 oznacza wariancję (również liczoną po czasie). Górny wykres na rysunku 3.9 przedstawia znormalizowaną³ liczbę linii genetycznych w funkcji czasu oraz funkcję rozrzutu $\rho(t)$. Jak można zauważyć, $\rho(t)$ osiąga jedno maksimum dla $500 < t < 1000$ po czym następuje jego monotoniczny spadek. Zachowanie $G(t)$ jest nam znane z poprzednich wykresów, zostało umieszczone tu w celach porównawczych. Dolny wykres przedstawia $\rho(t_0)$. Współczynnik korelacji, który jest stosunkowo niski gdy rozpatrujemy cały przedział czasowy, rośnie monotonicznie wraz ze wzrostem t_0 , zmierzając do 1. W miarę jak układ dąży do stanu stacjonarnego, te dwie wielkości stają się coraz bardziej skorelowane.

W celu pełniejszej oceny tego, jak te dwie wielkości – $\rho(t)$ i $G(t)$ wpływają na siebie, na rysunku 3.10 zestawiono zależność $\rho(G)$. Dla dużych wartości G

³przez znormalizowanie funkcji dyskretnej f rozumiemy fakt, że spośród wszystkich wartości funkcji została wybrana wartość największa f_{\max} i przez nią zostały podzielone wszystkie wartości funkcji.



Rysunek 3.10: Zależność funkcji rozrzutu od liczby linii genetycznych w populacji.

(początkowych kroków czasowych) jest to funkcja malejąca, zaś poniżej pewnej wartości krytycznej – rosnąca (większość punktów na prezentowanym wykresie).

Przedstawione tu rozważania pozwalają na wyciągnięcie wniosku, iż spadek różnorodności genomów w populacji związany jest ze zmniejszaniem się liczby linii genetycznych w populacji. Zwiększenie parametru P (współczynnik mutacji odwrotnych) zmniejsza występujące w funkcji $m(k)$ (zmieniając również charakter pozostałych zależności). Z przeprowadzonych dotąd przebiegów symulacji wynika, iż dla $P \approx 0.1 \cdot M$ na ogół zanikają one całkowicie, choć trudne jest wyciągnięcie w tej kwestii jednoznacznych wniosków.

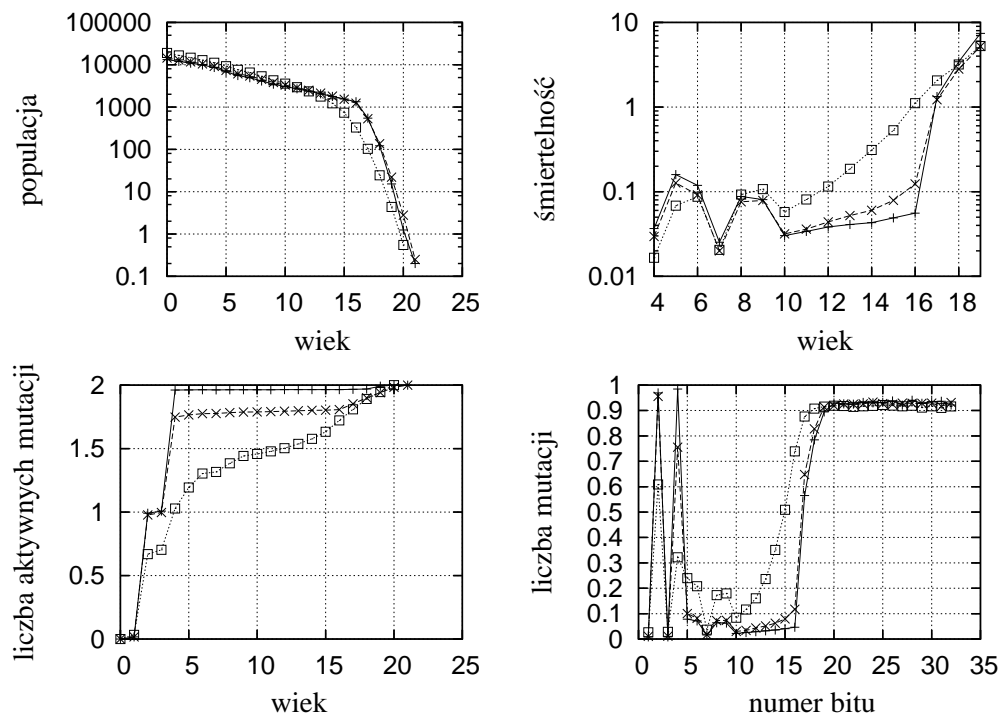
Wiele informacji na temat samego zjawiska spadku linii genetycznych, zawiera pozycja [24]. Jej autorzy dowodzą między innymi, że nieprzewidywalnym jest fakt, która linia genetyczna zostanie przy życiu (badając współzawodnictwo dwóch linii), a sam rozmiar pojedynczej linii podlega zmianom o charakterze błędzenia przypadkowego.

3.4 Rola selektywnego doboru partnera w modelu seksualnym

W dotychczasowych rozważaniach braliśmy pod uwagę jedynie model w którym dobór partnera następuje w sposób losowy. W tej części pracy wprowadzimy dodatkowe trzy parametry do modelu, które kontrolują sposób doboru partnera podczas rozmnażania, oraz zajmiemy się odpowiedzią na pytanie: w jaki sposób wprowadzenie preferencji w takim doborze, wpływa na strukturę populacji. Do modelu wprowadzamy jeden typ preferencji – preferencje na podstawie genomu osobników.

Jak wynika z przedstawionych wcześniej rezultatów, najbardziej istotne są początkowe bity genomu, gdyż one decydują o tym czy osobnik jest w stanie się rozmnożyć. Końcowe bity natomiast odgrywają znacznie mniejszą rolę, w większości przypadków osobnik nie dożywa wieku w którym zostaną one “odsłonięte”. W naszej modyfikacji modelu, pierwszy z wprowadzonych parametrów l , mówi ile początkowych bitów genomu jest istotnych dla selekcji partnera. Drugi parametr r określa możliwą odległość między genomami partnerów, w sensie odległości Hamminga (suma wartości bezwzględnych różnic pomiędzy bitami na poszczególnych pozycjach w genomach osobników). Trzeci z parametrów p – ile prób znalezienia partnera mogą wykonywać osobniki (przedstawione w pracy wyniki zostały uzyskane dla parametru $p = 4$). Zatem proces dobierania partnera odbywa się następująco:

- dla każdej losowo dobranej pary osobników (i, j) , obliczana jest odległość Hamminga pomiędzy jej genomami (ściślej ujmując, pomiędzy bitami z

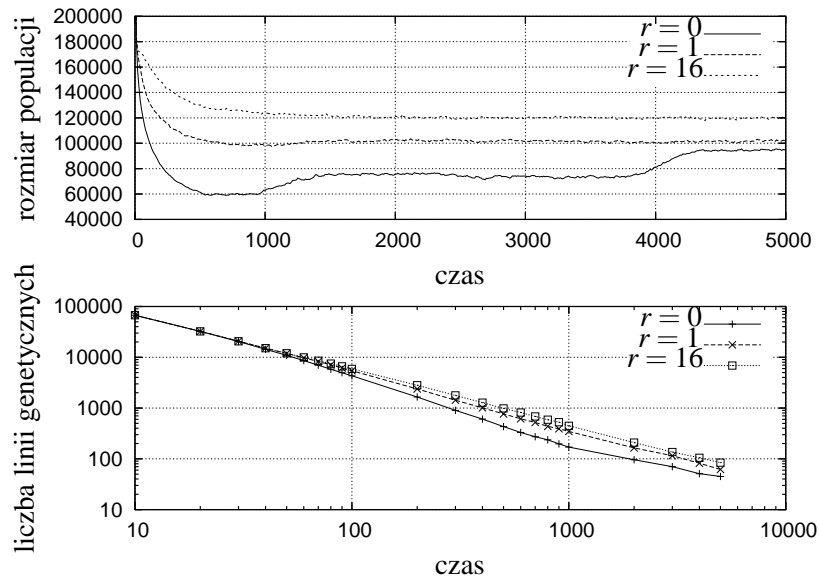


Rysunek 3.11: Porównanie struktury populacji – wyniki symulacji dla modelu płciowego z preferencją doboru partnera dla różnych wartości parametru r : $r = 0$ ('+'), $r = 1$ ('x'), oraz $r = 16$ ('□' – brak preferencji). Pozostałe parametry: $(N, B, R, M, P, T) = (10^6, 1.3, 7, 1, 0.1, 3)$, $l = 16$, $p = 4$.

chromosomów na pozycjach dominujących):

$$h(i, j) = \sum_{k=1}^l |g^*(i, k) - g^*(j, k)| \quad (3.12)$$

- jeśli $h \leq r$, wówczas dana para wydaje potomstwo w liczbie odpowiadającej parametrowi B .
- jeśli $h > r$, obydwa osobniki rozpoczynają następną próbę znalezienia partnera, aż liczba prób osiągnie p .



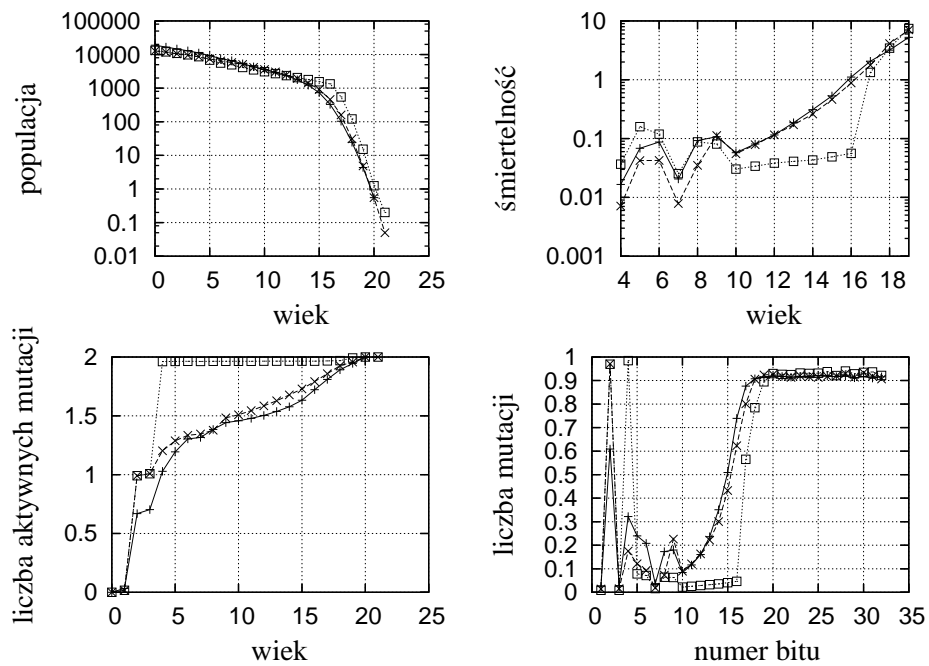
Rysunek 3.12: Porównanie wyników symulacji ($n(t)$ i $G(t)$) dla modelu płciowego z preferencją doboru partnera dla różnych wartości parametru r .

Wykresy na rysunkach 3.11 i 3.12 przedstawiają uzyskane wyniki symulacji (5000 kroków) dla modelu seksualnego z preferencją w doborze partnera, przy różnych wartościach parametru $r \in \{0, 1, 16\}$. Przy obliczaniu odległości między genomami brana pod uwagę była tylko pierwsza połowa genomu ($l = 16$). Wynik dla $r = 16$ odpowiada modelowi bez wprowadzonych preferencji i jest analogiczny do prezentowanego we wcześniejszej części pracy, natomiast $r = 0$ oznacza, że osobniki aby utworzyć parę, muszą mieć identycznych pierwszych $l = 16$ bitów w chromosomach na pozycjach dominujących.

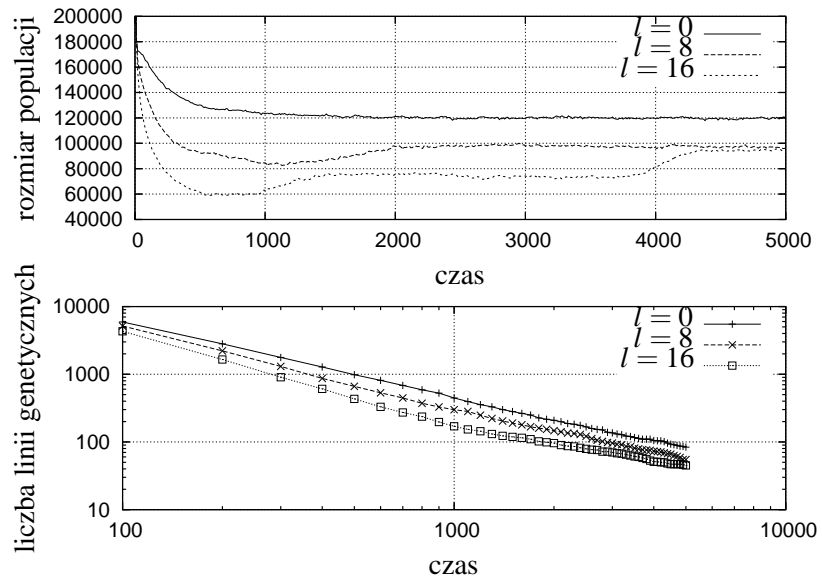
W przypadku gdy $r = 0$, rosną gwałtownie maksima w funkcji $m(k)$, co powoduje również, iż $\mu(a)$ zmienia się skokowo. Dla $r = 16$ mamy wyższą śmiertelność dla $a > 10$ w porównaniu z przypadkiem $r = 0$. Jednak dla osobników

najmłodszych sytuacja ta odwraca się, a ponieważ jak możemy zauważyć z zależności $n(a)$ (lewy górny wykres na rys. 3.11) jest ich znacznie więcej, skutkuje to tym, iż całkowita liczba osobników jest nieco większa w przypadku $r = 16$ (górny wykres na rysunku 3.12). Zależność $G(t)$ (dolny wykres na rysunku 3.12) wygląda dla wszystkich trzech przypadków podobnie, choć dla $r = 0$ odbiega ona nieznacznie od zależności potęgowej. Wystarczy jednak zwiększenie r o 1 by zależność ta stała się niemal identyczna jak dla przypadku bez preferencji w doborze partnera ($r = 16$). Również zależność rozmiaru populacji od czasu $n(t)$, jest nieco inna w przypadku gdy $r = 0$ – pojawiają się charakterystyczne “przeskoki”. O ile początkowe wahania $n(t)$ dla tego przypadku nie są niczym nowym, gdyż układ jeszcze nie osiągnął stanu stacjonarnego, to zaskakujący jest fakt, iż następuje wzrost $n(t)$ w pobliżu $t \approx 4000$, gdy mogłoby się wydawać, że po tak długim czasie utrzymywania się $n(t)$ na stałym poziomie nie ulegnie ona zmianie. Wydaje się, iż przyczyną tego może być to, że populacja stała się już na tyle jednorodna, że osobniki łatwiej znajdują partnera, dzięki czemu liczba narodzin rośnie. Sytuacja ta skłania do ostrożnego formułowania wniosków na temat tego czy układ osiągnął stan stacjonarny.

Wykresy na rysunkach 3.13 i 3.14 przedstawiają analogiczne porównanie wyników, lecz dla różnych wartości parametru $l \in \{0, 8, 16\}$. Dla każdego z przypadków $r = 0$, zatem model bez preferencji dostajemy dla $l = 0$. Gdy porównamy zachowanie się $m(k)$ w obszarze pierwszych ośmiu bitów, zauważymy że rezultaty są zbliżone do siebie dla przypadków z $l = 8$ i $l = 16$, zaś dla następnych ośmiu bitów – zbliżone rezultaty mamy dla $l = 8$ i $l = 0$. Sytuację tę nieco łatwiej zauważyć w zachowaniu $\mu(a)$. Największy rozmiar populacji otrzymujemy zgodnie z przewidywaniami dla wersji bez preferencji $l = 0$, mniejszy dla $l = 8$ i $l = 16$ – im więcej bitów potencjalnych partnerów musi do siebie pasować, tym mniej powstaje par w populacji i stąd mniejsza całkowita liczba narodzin. Zależność $G(t)$ nie podlega większym zmianom wraz ze zmianą l , obserwujemy jedynie nieco większą liczbę linii genetycznych dla odpowiednio większej populacji.



Rysunek 3.13: Porównanie struktury populacji – wyniki symulacji dla modelu płciowego z preferencją doboru partnera: $l = 0$ ('+') oznacza brak preferencji, zaś $l = 8$ ('x') i $l = 16$ ('□') – oznaczają iż osobniki krzyżują się tylko z tymi które posiadają identycznych odpowiednio 8 i 16 pierwszych bitów (dominujących) w genomie



Rysunek 3.14: Porównanie zależności $n(t)$ i $G(t)$ dla modelu płciowego z preferencją doboru partnera, dla różnych wartości parametru $l \in \{0, 8, 16\}$.

Rozdział 4

Wnioski i podsumowanie

Przedstawione w pracy wyniki i wnioski, pozwalają na podsumowanie celów postawionych we wstępnej części pracy.

W podrozdziale 3.3 przedstawiliśmy zasadniczą różnicę w sposobie w jaki zmienia się liczba linii genetycznych w czasie, dla różnych wersji modelu. Dla modelu seksualnego zależność ta: $G(t) = a \cdot t^u$, jest taka sama na całym obserwowanym przedziale – od początku symulacji, aż do chwili gdy liczba linii osiągnie wartość równą 1. Przy czym stała $u \approx -1$ nie zależy od parametrów symulacji. W standardowym modelu Penny natomiast, co prawda dynamika zaniku linii genetycznych ma ten sam charakter $G(t) = a \cdot t^u$, lecz $u \approx -1$ jedynie przez kilkaset pierwszych kroków czasowych. Potem zależność ta przechodzi w obszar gdzie $u \approx -2$ i pozostaje w tym obszarze praktycznie do końca badanego czasu symulacji, czyli aż pozostanie w populacji tylko jedna linia genetyczna. Zatem podsumowując: zależność $G(t)$ dla obu wersji ma ten sam charakter, lecz dla wersji seksualnej $u \approx -1$, zaś dla aseksualnej w przeważającej części $u \approx -2$. Wynik ten odbiega od przedstawianego dotychczas w literaturze, gdzie niedostatecznie wysoka pojemność środowiska uniemożliwiała zaobserwowanie zjawiska.

W podrozdziale 3.2 podzieliliśmy śmiertelność na dwa składniki: składnik pochodzący od czynnika Verhulsta, oraz składnik genetyczny (śmiertelność z powodu nadmiaru aktywnych mutacji). Jak zostało to pokazane na przykładowych wynikach symulacji, jedynie składnik genetyczny daje wykładniczy wzrost śmiertelności wraz z wiekiem w całym przedziale wiekowym. Problem ten nie jest co prawda nowy i na ogół tak zdefiniowana śmiertelność jest podstawą analizy danych z symulacji. Nie został on jednak dotąd postawiony w ten sposób w dostępnej nam literaturze, a zasługuje na oddzielne opisanie.

Przedstawiliśmy również w podrozdziale 3.4, opis i analizę wyników zmodyfikowanego seksualnego modelu Penny. W modelu tym genom osobnika ma wpływ na dobór partnera. Upraszczając, szansę na stworzenie pary mają tylko te osobniki, dla których odległość Hamminga pomiędzy słowami bitowymi repre-

zientującymi ich genomy jest nie większa niż zadany parametr r . Dla wersji bez preferencji dostajemy największy rozmiar populacji, natomiast im “silniejsze” są preferencje doboru partnera (im mniejsze jest r), tym bardziej rozmiar populacji spada. Położenie $r = 0$ (“najsilniejsze” preferencje) powoduje również, iż zależność $G(t)$ nieznacznie odbiega od $G(t) = a \cdot t^u$.

Wymienione powyżej rezultaty, nasuwają na myśl możliwości dalszych badań. W pracy nie zajmowaliśmy się w ogóle problemem różnic między płciami, takimi jak choćby występowanie chromosomów XX i XY (omówienie tych różnic na bazie modelu Penny zawierają między innymi prace [22] oraz [14]). Dlatego też jednym z kierunków przyszłych badań, mogła by być wersja która zarówno uwzględnia preferencje w doborze partnera, jak i wspomniane różnice pomiędzy osobnikami różnych płci.

Również omówiony problem charakteru funkcji $G(t)$ zostawia pole do przyszłych badań. Nie jest wykluczone, iż dla dużo większych pojemności środowiska N nie pojawi się jeszcze jedna zmiana parametru u . Problem ten został zbadany w granicach na które pozwalają nam rozmiary pamięci operacyjnej w dostępnych nam maszynach.

Ciekawym przedmiotem badań, mogła by być jedna z wersji modelu Penny bez czynnika Verhulsta, jak choćby jedna z opisanych w [34], [35], lub [36]. W szczególności interesująca wydaje się możliwość zbadania charakteru funkcji $G(t)$ w zależności od przyjętego modelu oraz możliwość przeniesienia do wymienionych wersji, opisanych preferencji w doborze partnera.

Dodatek A

Użyte oznaczenia

W pracy użyto następujących oznaczeń:

- $n(a, t)$ – liczba osobników w wieku a w chwili czasowej t ;
- $n(a)$ – liczba osobników w wieku a , gdy nie zależy ona od czasu;
- $n(t)$ – całkowita liczba osobników w populacji, w chwili czasowej t ;
- $s(a)$ – współczynnik przetrwania dla osobników w wieku a ;
- $q(a)$ – współczynnik śmiertelności dla osobników w wieku a ;
- s – współczynnik przetrwania, gdy nie zależy on od wieku osobnika;
- q – współczynnik śmiertelności, gdy nie zależy on od wieku osobnika;
- $g_i(k), g(i, k)$ – wartość genomu i -tego osobnika na k -tej pozycji, niezależna od czasu;
- $g^*(i, k)$ – wartość genomu i -tego osobnika na dominującej k -tej pozycji (model seksualny), niezależna od czasu;
- $g_t(i, k)$ – wartość genomu i -tego osobnika na k -tej pozycji w kroku czasowym t ;
- $\mu_i(a)$ – liczba aktywnych mutacji i -tego osobnika w wieku a ;
- $m(k) = \frac{1}{n} \sum_{i=1}^n g_i(k)$ – funkcja rozkładu mutacji;
- $\overline{\mu(a)} = \frac{1}{n} \sum_{i=1}^n \mu(a, i)$ – liczba aktywnych mutacji uśredniona po wszystkich osobnikach w populacji;

- $k_v(t, a)$ – liczba przypadków śmierci osobników w wieku a ; w kroku czasowym t , z powodu czynnika Verhulsta;
- $k_v(a)$ – liczba przypadków śmierci osobników w wieku a , z powodu czynnika Verhulsta, gdy nie zależy ona od czasu;
- $k_m(t, a)$ – liczba przypadków śmierci osobników w wieku a , w kroku czasowym t , z powodu nadmiaru aktywnych mutacji;
- $k_m(a)$ – liczba przypadków śmierci osobników w wieku a , z powodu nadmiaru aktywnych mutacji, gdy nie zależy ona od czasu;
- $q_m(a) = \frac{k_m(a)}{n(a)}$ – śmiertelność powodowana mutacjami (niezależna od czasu);
- $\overline{g(k)} = \frac{1}{n} \sum_{i=1}^n g(i, k)$ – średnia arytmetyczna po genomach wszystkich osobników na pozycji k ;
- $\sigma^2(k) = \frac{1}{n} \sum_{i=1}^n \left(g(i, k) - \overline{g(k)} \right)^2$ – wariancja z wartości bitu na pozycji k liczona po wszystkich osobnikach;
- $\rho(t) = \overline{\sigma_t^2(k)} = \frac{1}{32} \sum_{k=1}^{32} \sigma_t^2(k)$ – funkcja rozrzutu mutacji;
- $\overline{\rho(t_0)}$ – średnia arytmetyczna z funkcji rozrzutu, liczona po wszystkich krokach czasowych, począwszy od zadanego t_0 do ostatniego kroku czasowego symulacji;
- $\overline{G(t_0)}$ – średnia arytmetyczna z liczby linii genetycznych w populacji, liczona po wszystkich krokach czasowych, począwszy od zadanego t_0 do ostatniego kroku czasowego symulacji;
- $\overline{\rho(t_0)G(t_0)}$ – średnia arytmetyczna z iloczynów funkcji rozrzutu i liczby linii genetycznych, liczona po wszystkich krokach czasowych, począwszy od zadanego t_0 do ostatniego kroku czasowego symulacji;
- $\sigma_\rho^2(t_0)$ – wariancja z funkcji rozrzutu, liczona po wszystkich krokach czasowych, począwszy od zadanego t_0 do ostatniego kroku czasowego symulacji;
- $\sigma_G^2(t_0)$ – wariancja z liczby linii genetycznych, liczona po wszystkich krokach czasowych, począwszy od zadanego t_0 do ostatniego kroku czasowego symulacji;

Dodatek B

Kluczowe fragmenty kodu źródłowego programu

Poniżej zostały zamieszczone dwa główne fragmenty kodów źródłowych programu napisanego przez autora i użytego do przeprowadzenia omawianych w pracy symulacji. Klasa o nazwie “Individual” reprezentuje pojedynczego osobnika (jego genom, wiek, liczbę aktywnych mutacji), implementuje symulację mechanizmów jego śmierci, oraz narodziny potomka.

Klasa o nazwie “Environment” reprezentuje całe środowisko w którym umieszczone zostają osobniki. Zawiera funkcję służącą do inicjowania środowiska (wypełniania osobnikami na początku symulacji), oraz funkcję realizującą pojedynczy krok czasowy symulacji.

Listing B.1: Kod źródłowy klasy “Individual”

```
import java.io.*;
import java.util.BitSet;

/**
5  * Represents one individual
  * */

public class Individual implements Serializable {
    private int age;
10  private int activeMutations;
    private BitSet genome;
    private int name;

    /** Initialize individual
15  * @param genomeLength length of the genome */
```

DODATEK B. KLUCZOWE FRAGMENTY KODU ŹRÓDŁOWEGO
PROGRAMU

```
    public Individual(int genomeLength) {
        genome = new BitSet(genomeLength);
    }
20
    /** Initialize individual
     * @param genomeLength length of the genome
     * @param number number of individual */

25    public Individual(int genomeLength,int name) {
        this(genomeLength);
        this.name=name;
    }

30    /** Initialize individual
     * @param genome initial genome */
    public Individual(BitSet genome){
        this.genome = (BitSet) genome.clone();
    }

35    /** Initialize individual
     * @param genome initial genome
     * @param number number of individual */
    public Individual(BitSet genome,int name){
40        this.genome = (BitSet) genome.clone();
        this.name=name;
    }

    /** Check if the individual is alive
45    */
    public boolean isAlive(RuntimeParameters rt, Parameters par){

        // genetic death
        if(activeMutations >= par.getT()) {
50            rt.mutationKilledIndividuals++;
            rt.mutationKilledTable[age]++;
            return false;
        }

55    // verhulst death
        if(Extern.rnd.nextDouble() < rt.verhulstFactor){
            rt.verhKilledIndividuals++;
            rt.verhKilledTable[age]++;
            return false;
        }
    }
```

DODATEK B. KLUCZOWE FRAGMENTY KODU ŹRÓDŁOWEGO
PROGRAMU

```
60     }

    // hunting death
    if(par.getH() > 0.0)
        if(Extern.rnd.nextDouble() < par.getH()) {
65             rt.huntedIndividuals++;
            return false;
        }

    // end of genome (death)
70    if(age >= par.getGenomeLength()) {
        rt.xxKilledIndividuals++;
        return false;
    }

75    return true;
}

/** Make individual older by one unit
 * */
80 public void makeOlder(){
    if(getGenome(++age))
        activeMutations++;
}

85

/** return list of born individuals
 * by actual parent according to actual birth rate
 * */
public IndividualsList giveBirth(RuntimeParameters rt,Parameters par) {
90     IndividualsList born = new IndividualsList();
    Individual bi;

    if(age >= par.getR()) {
        for(int i=1;i<=(int)par.getB();i++) {
95             bi = new Individual(par.getGenomeLength());
            bi.inheritGenome(genome,par);
            bi.name=this.name;
            born.add(bi);
            rt.bornIndividuals++;
100        }

        float bProb = par.getB() - (int)par.getB();
```

DODATEK B. KLUCZOWE FRAGMENTY KODU ŹRÓDŁOWEGO
PROGRAMU

```

        if( bProb > 0 && Extern.rnd.nextDouble() < bProb ){
            bi = new Individual(par.getGenomeLength());
105         bi.inheritGenome(genome,par);
            bi.name=this.name;
            born.add(bi);
            rt.bornIndividuals++;
        }
110     }
    return born;
}

115  /** Inherit genome from parent with added positive and negative mutations
    * @param parentGenome genome inherited from parent
    * @param M probability of giving additional "bad" mutations,
    * that means: <tt>n</tt> mutations and one with probability <tt>pn</tt>,
    * where <tt>n</tt> is an integer part of M, and
120  * <tt>pn</tt> is a mantissa of it.
    * @param P probability of giving additional
    * "positive" mutations
    * */

125  public void inheritGenome(BitSet parentGenome,Parameters par){

        Rand rnd = Extern.rnd;

        genome = (BitSet) parentGenome.clone();

130     // negative mutations
        if(par.getM() > 0.0) {
            for(int i=1;i<=(int)par.getM();i++){
                int position = rnd.nextInt(genome.size() - 1);
135         genome.set(position,true);
            }
            if(rnd.nextDouble() < par.getM()-(int)par.getM()){
                int position = rnd.nextInt(genome.size() - 1);
                genome.set(position,true);
140         }
        }

        // positive mutations
145     if(par.getP() > 0.0) {
```


DODATEK B. KLUCZOWE FRAGMENTY KODU ŹRÓDŁOWEGO
PROGRAMU

```
        for(int i=1;i<=(int)par.getP();i++){
            int position = rnd.nextInt(genome.size() - 1);
            genome.set(position,false);
        }
150    if(rnd.nextDouble() < par.getP()-(int)par.getP()){
        int position = rnd.nextInt(genome.size() - 1);
        genome.set(position,false);
    }
    }
155 }

/**
 * Get genome element at specified index.
 *
160 * @param index the index.
 * @return genome at index as boolean.
 */
public boolean getGenome(int index) {
    // genome is shifted: age=0 hasn't his position
165    return genome.get(index-1);
}

/**
 * Set genome at the specified index.
170 *
 * @param index the position.
 * @param state state to set
 */
public void setGenome(int index,boolean state) {
175    // genome is shifted: age=0 hasn't his position
    genome.set(index-1,state);
    //this.genome[index-1] = state;
}

180

/**
 * Get age.
 *
185 * @return age as int.
 */
public int getAge() {
    return age;
}
```

DODATEK B. KLUCZOWE FRAGMENTY KODU ŹRÓDŁOWEGO PROGRAMU

```
190    /**
        * Set age.
        *
        * @param age the value to set.
        */
195    public void setAge(int age) {
        this.age = age;
    }

    /**
200    * Get activeMutations.
        *
        * @return activeMutations as int.
        */
    public int getActiveMutations() {
205        return activeMutations;
    }

    /**
        * Set activeMutations.
210    *
        * @param activeMutations the value to set.
        */
    public void setActiveMutations(int activeMutations) {
215        this.activeMutations = activeMutations;
    }

    /**
        * Get genome.
        *
220    * @return genome as BitSet
        */
    public BitSet getGenome() {
        return genome;
    }
225

    /**
        * Set genome.
230    *
        * @param genome the value to set.
```

DODATEK B. KLUCZOWE FRAGMENTY KODU ŹRÓDŁOWEGO
PROGRAMU

```
    */
    public void setGenome(BitSet genome) {
        this.genome = genome;
235    }
```

```
    /**
240     * Get name.
     *
     * @return name as int.
     */
    public int getName()
245    {
        return name;
    }
}
```

Listing B.2: Kod źródłowy klasy “Environment”

```
import java.util.*;

/** Represents environment where individuals live
5  */
public class Environment implements Constants {

    private IndividualsList ind; // list of individuals
    private IndividualsIterator it; // iterator from above list
10    private RuntimeParameters rt; // variables that represents actual state
    private Parameters par; // parameters of the simulation

    /** Initialize environment
15     */
    public Environment(Parameters par) {
        ind = new IndividualsList();
        rt = new RuntimeParameters();
        this.par=par;
20    }

    /** Fill environment
     * @param genomeType type of genome
     */
}
```

```
25  public void fill() throws InputFileParseException {
    int genomeLength=par.getGenomeLength();
    int initialPopulation=par.getInitialPopulation();
    int name=1;
    Rand rnd;

30  switch(par.getInitGenomeType()) {
    case GENOME_CLEAN:
        for(int i=0;i<par.getInitialPopulation();i++)
            ind.add(new Individual(genomeLength,name++));
        break;
    case GENOME_HALF:
        rnd = Extern.rnd;
        BitSet genome = new BitSet(genomeLength);
        for(int i=0;i<initialPopulation;i++){
40            for(int j=0;j<genome.size();j++)
                genome.set(j,(rnd.nextInt(1)==1)?true:false);
            ind.add(new Individual(genome,name++));
        }
        break;
    case GENOME_PROB:
45        rnd = Extern.rnd;
        genome = new BitSet(genomeLength);
        for(int i=0;i<initialPopulation;i++){
            for(int j=0;j<genome.size();j++)
50                genome.set(j,(rnd.nextDouble()
                    < par.getInitialMutProbability())?true:false);
            ind.add(new Individual(genome,name++));
        }
        break;
    case GENOME_TABLE:
55        rnd = Extern.rnd;
        genome = new BitSet(genomeLength);
        for(int i=0;i<initialPopulation;i++){
            for(int j=0;j<genome.size();j++)
60                if(rnd.nextDouble() < par.initMutationsTable[j])
                    genome.set(j,true);
                else
                    genome.set(j,false);
            ind.add(new Individual(genome,name++));
65        }
        break;
    default:
```

*DODATEK B. KLUCZOWE FRAGMENTY KODU ŹRÓDŁOWEGO
PROGRAMU*

```
        throw new InputFileParseException("unknown_genome_type");
    }
70 }

    /** Run one time step of simulation
    */
    public void oneTimeStep() throws PopulationExtinctException {
75
        IndividualsList born = new IndividualsList();
        it = new IndividualsIterator(ind.getCollection());

        rt.verhulstFactor=(float)ind.size()/(float)par.getN();
80 rt.bornIndividuals=0;
        rt.verhKilledIndividuals=0;
        rt.mutationKilledIndividuals=0;
        rt.huntedIndividuals=0;
        rt.xxKilledIndividuals=0;
85

        rt.mutationKilledTable=new int[par.getGenomeLength()+1];
        rt.verhKilledTable=new int[par.getGenomeLength()+1];
        Arrays.fill(rt.mutationKilledTable,0);
        Arrays.fill(rt.verhKilledTable,0);
90

        rt.timeStep++; // go one step forward

        // begining of list is an exception
        Individual current = it.next();
95 current.makeOlder();
        while(!current.isAlive(rt,par)) {
            it.remove();
            if(!it.hasNext())
                throw new PopulationExtinctException(rt.timeStep);
100        else {
            current = it.next();
            current.makeOlder();
        }
    }
105 born.addAll(current.giveBirth(rt,par));

    while(it.hasNext()){
        current = it.next();
        current.makeOlder();
110        if(current.isAlive(rt,par))
```

DODATEK B. KLUCZOWE FRAGMENTY KODU ŹRÓDŁOWEGO PROGRAMU

```
        born.addAll(current.giveBirth(rt,par));
    else
        it.remove();
    }
115
    // connect two lists
    ind.addAll(born);
}

120
// Begin: getters, setters

/**
 * Get individuals.
125  *
 * @return individuals as List.
 */
public IndividualsList getIndividuals() {
    return ind;
130 }

/**
 * Set individuals.
 *
135 * @param individuals the value to set.
 */
public void setInd(IndividualsList individuals) {
    this.ind = individuals;
}

140
// End: getters, setters

/**
145 * Get ind.
 *
 * @return ind as IndividualsList.
 */
public IndividualsList getInd()
150 {
    return ind;
}
```

DODATEK B. KLUCZOWE FRAGMENTY KODU ŹRÓDŁOWEGO
PROGRAMU

```
155     /**
        * Get rt.
        *
        * @return rt as RuntimeParameters.
        */
    public RuntimeParameters getRt()
160    {
        return rt;
    }

    /**
165     * Get par.
        *
        * @return par as Parameters.
        */
    public Parameters getPar()
170    {
        return par;
    }

    public void setRt(RuntimeParameters rt){
175        this.rt=rt;
    }

    public void setPar(Parameters par){
        this.par=par;
180    }
}
```

Bibliografia

- [1] <http://www.mortality.org>. *The Human Mortality Database* (2003).
- [2] T.J.P. Penna. *A Bit-String Model for Biological Ageing*. *J. Stat. Phys.*, 72(1629) (1995).
- [3] Leonard Hayflick. *Jak i dlaczego się starzejemy*. Książka i wiedza (1998).
- [4] Gustavo Barja. *Trends in Neurosciences*, 27(10) (2004).
- [5] Toupance B., Godelle B., Gouyon P.H., i Schachter F. *American Journal of Human Genetics*, 6(62) (1998).
- [6] R. Wieczorkowski i R. Zieliński. *Komputerowe generatory liczb losowych*. WNT (1997).
- [7] W. Press, S. Teukolsky, W. Vetterling, i B. Flannery. *Numerical Recipes in C*. Cambridge University Press.
- [8] H. G. Schuster. *Chaos deterministyczny*. Wydawnictwo Naukowe PWN (1993).
- [9] <http://www.prb.org>. *Population Reference Bureau* (2003).
- [10] <http://www.stat.gov.pl>. *Polska Statystyka Publiczna* (2003).
- [11] M. Sitarz. *Population structure in the Penna model*. *Task Quarterly*, 7 263–269 (2003).
- [12] B. Gompertz. *On the Nature of the Function Expressive of the Law of Human Mortality, and on a New Mode of Determining the Value of Life Contingencies*. *Philosophical Transactions of the Royal Society of London*, 115 (1825).
- [13] S. Moss de Oliveira, Paulo M. C. Oliveira, D. Stauffer, i S. Moss de Oliveira. *Evolution, Money, War, and Computers*. Teubner Verlag (1999).

- [14] J. Schneider, S. Cebrat, i D. Stauffer. *why do women live longer than men? A monte carlo simulation of Penna-type models with X and Y chromosomes. International Journal of Modern Physics C*, 9 721–725 (1998).
- [15] D. Stauffer, P.M.C. de Oliveira, i S. Moss de Oliveira R.M. Zorzenon dos Santos. *Monte Carlo simulations of sexual reproduction. Physica A*, (231) 504–514 (1996).
- [16] A. Pękański. *Model of evolution with sexual and non-sexual reproduction. The European Physical Journal B*, (13) 791–796 (1999).
- [17] M. S. Magdoń-Maksymowicz, M. Sitarz, M. Bubak, A.Z. Maksymowicz, i J. Szewczyk. *Effect of time slowing in biological ageing. Computer Physics Communications*, 147 621–624 (2002).
- [18] D. Makowiec. *Penna model of biological ageing on a lattice. Acta Physica Polonica B*, 31 1037–1049 (2000).
- [19] A. Z. Maksymowicz. *Influence of variations in threshold of bad mutations on age structure of the population. Physica A*, 237 150–157 (1999).
- [20] Maria S. Magdoń i Andrzej Z. Maksymowicz. *Penna model in migrating population – effect of environmental factor and genetics. Physica A*, (273) 182–189 (1999).
- [21] Paul Berg i Maxine Singer. *Język genów. Poznawanie zasad dziedziczenia. Prószyński i S-ka* (1997).
- [22] S. Cebrat. *Penna model from the perspective of one geneticist. Physica A* 258, 493–498 (1998).
- [23] http://en.wikipedia.org/wiki/Mitochondrial_Eve. *Wikipedia – Mitochondrial Eve* (2004).
- [24] D. Makowiec, J. Dąbkowski, i M. Groth. *The Eve effect in the Penna model of biological aging. Physica A*, 273(159) (1999).
- [25] T. Ozawa. *Genetic and functional changes in mitochondria associated with aging. Physiological Reviews*, 77 425–464 (1997).
- [26] <http://www.cdc.gov/nchs/>. *National Center for Health Statistics* (2004).
- [27] M. E. Cates J. B. Coe, Y. Mao. *Solvable Senescence Model Showing a Mortality Plateau. Physical Review Letters*, 89 (2002).

- [28] <http://java.sun.com>. *Java Technology* (2004).
- [29] <http://www.answers.com/topic/just-in-time-compilation>. *Just-in-time compilation* (2005).
- [30] T. Suganuma, T. Ogasawara, M. Takeuchi, T. Yasue, M. Kawahito, K. Ishizaki, H. Komatsu, , i T. Nakatani. *Overview of the IBM Java Just-in-Time Compiler*. *IBM Systems Journal*, 39(1) (2000).
- [31] <http://gcc.gnu.org/java/>. *GNU Java Compiler* (2005).
- [32] K. Malarz. *Searching for scaling in the Penna bit-string model of biological aging*. *International Journal of Modern Physics C*, 11 309–315 (2000).
- [33] M. Sitarz i A. Z. Maksymowicz. *International Journal of Modern Physics C*, 16(12) (2006).
- [34] J. Dabkowski, M. Groth, i D. Makowiec. *Acta Phys. Polonica*, 31(1027) (2000).
- [35] S.M. de Oliveira, P.M.C de Oliveira, i J.S.Sa. Martins. *International Journal of Modern Physics C*, 15 (2003).
- [36] J.S. Martins i S. Cebrat. *Theory in Biosciences*, 119(156) (2000).