Bioinformatyka

wykład 6: asemblacja

prof. dr hab. inż. **Marta Kasprzak**
Instytut Informatyki, Politechnika Poznańska

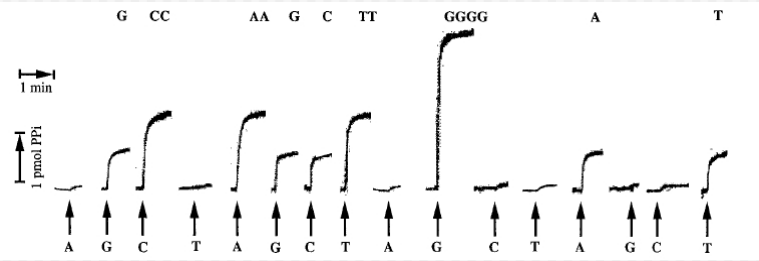
Sekwencjonowanie nowej generacji

[M. Ronaghi i in., *Anal. Biochem.* 242 (1996)]

- Pirosekwencjonowanie to biochemiczna metoda sekwencjonowania DNA polegająca na odczycie fragmentu pojedynczej nici DNA poprzez syntezę wzdłuż tego fragmentu jego komplementarnego odpowiednika. Na jej podstawie oparto po latach działanie pierwszego zautomatyzowanego sekwenatora
- W trakcie tej syntezy, za każdym razem gdy kolejny nukleotyd jest dodawany do nici komplementarnej, uruchamiana jest seria reakcji chemicznych kończących się emisją światła

Sekwencjonowanie nowej generacji

[M. Ronaghi i in., *Anal. Biochem.* 242 (1996)] – cd.



- Ilość światła zależy od liczby nukleotydów tego samego typu dodanych w jednym kroku. Po każdym kroku reszta nieprzyłączonych nukleotydów jest wypłukiwana z roztworu i wprowadzane są kolejne innego typu

Sekwencjonowanie nowej generacji

[M. Margulies i inni, *Nature* 437 (2005)]

- Firma 454 Life Sciences (później w Roche) zastosowała pirosekwencjonowanie do asemblacji DNA. Cały proces sekwencjonowania odbywa się maszynowo, metody algorytmiczne stosuje się dopiero na etapie asemblacji
- Sekwenator 454 umożliwia odczytanie milionów nukleotydów w jednym przebiegu maszyny. Jest to historycznie pierwsza metoda szybkiego sekwencjonowania
- W wyniku sekwencjonowania otrzymywany jest zbiór sekwencji (o długości nawet ok. 700 nukleotydów) o wysokiej jakości odczytu i stosunkowo dużym pokryciu badanego fragmentu DNA (ok. 30−40 sekwencji na jedną pozycję we fragmencie). Jako dane uzupełniające otrzymuje się „wiarygodność” dla każdego odczytanego nukleotydu

Sekwencjonowanie nowej generacji

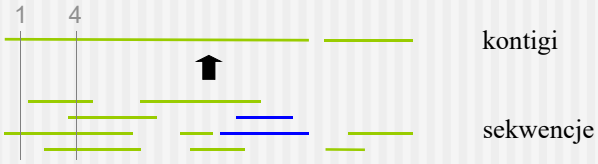
- Inne popularne technologie sekwencjonowania drugiej generacji: Solexa/Illumina (Illumina), SOLiD (Applied Biosystems)
- Wysoka jakość odczytywanych sekwencji oraz duże pokrycie nimi badanego fragmentu sprzyja sekwencjonowaniu znacznych fragmentów DNA, nawet całych genomów
- Obecnie standardem są protokoły dla odczytów sparowanych. Aby maksymalnie wykorzystać naturę tych danych, a także załączony do nich współczynnik wiarygodności nukleotydów, opracowywane są nowe algorytmy specjalizowane
- Rozwijane są technologie produkcji długich odczytów (PacBio, Oxford Nanopore, ponad 10/100 tys. nukleotydów, nawet miliony), gdzie największym wyzwaniem jest redukcja błędów sekwencjonowania

Asemblacja

- Asemblacja *de novo* łańcuchów DNA jest problemem trudnym obliczeniowo. Nawet bardzo ograniczony wariant tego problemu – problem najkrótszego wspólnego superciągu (ang. *shortest common superstring*) – jest silnie NP-trudny
- Problem asemblacji jest dodatkowo skomplikowany przez liczbę i różnorodność błędów występujących w instancji, a także przez jej znaczny rozmiar
- Sekwencje mogą pochodzić z obu nici DNA i ich orientacja nie jest znana. Zawierają przekłamania przeniesione z etapu sekwencjonowania: insercje, delecje i zamiany nukleotydów

Asemblacja

Cechy instancji problemu asemblacji DNA



- Ogólne sformułowanie problemu asemblacji zakłada sekwencje wejściowe różnej długości, gdzie jedne mogą zawierać się w innych
- Rozkład sekwencji w badanym fragmencie genomu jest nierównomierny. Brak pokrycia jest jedną z przyczyn rekonstrukcji genomu w postaci rozłącznych odcinków (tzw. *kontigów*)

Asemblacja

Fragmenty składowe:

```
TATGC, ATCAGCAAC, GACTC, GTAGA, GCAGCA
```

Jedno z możliwych rozwiązań:

```
TATGCAGCA-CTCTAC
```

```
TATGC G-A-CTC
   GCAGCA  TCTAC
 AT-CAGCAAC
```

Asemblacja

- Sformułowanie wersji optymalizacyjnej problemu asemblacji

Instancja: Multizbiór *S* sekwencji pochodzących z obu nici badanego łańcucha DNA.

Odpowiedź: Sekwencja wynikowa o maksymalnej wiarygodności zawierająca, z dopuszczonym pewnym odsetkiem niezgodności, wszystkie sekwencje z *S* czytane wprost lub z założeniem przeciwnej orientacji.

- Najczęściej rozwiązaniem jest nie jedna spójna sekwencja, lecz zbiór rozłącznych kontigów
- Istnieją alternatywne sformułowania dla tego problemu: inne kryterium optymalizacji (np. minimalizacja długości sekwencji wynikowej), inne ograniczenia (nieużycie części zbioru *S*)

Dopasowanie–uszeregowanie–konsensus

Dawniej algorytmy asemblacji opierały się często na trzyetapowym modelu obliczeń (ang. *overlap-layout-consensus*):

- Dopasowanie par sekwencji wejściowych – w celu znalezienia potencjalnych sąsiadów w rozwiązaniu, z dopuszczeniem pewnych niezgodności (dopasowanie semiglobalne)
- Uszeregowanie sekwencji – znajdowanie prawdopodobnego uporządkowania ich w sekwencji oryginalnej
- Generowanie sekwencji konsensusowej – wywiedzenie sekwencji nukleotydów ze znalezionego uszeregowania

Asemblacja – graf nałożeń

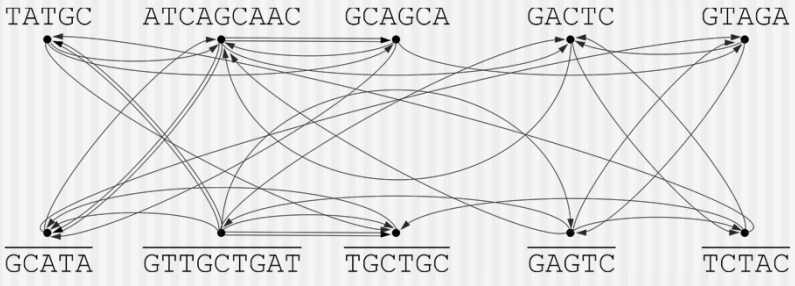
[J.D. Kececioglu i E.W. Myers, *Algorithmica* 13 (1995)]

- Na wejściu sekwencje z błędami (insercje, delecje, zamiany) pochodzące z obu nici łańcucha DNA. Zastosowano model obliczeń *overlap-layout-consensus*
- Porównanie wszystkich par sekwencji, z uwzględnieniem ich odwrotnie komplementarnych odpowiedników, w celu otrzymania nałożeń o dopuszczonym odsetku błędów
- Konstrukcja grafu skierowanego z wierzchołkami odpowiadającymi sekwencjom (×2) i łukami odpowiadającymi najlepszym nałożeniom danej pary wierzchołków
- Stosowane są różne oznaczenia łuków dla par sekwencji zazębiających się i dla takich, w których jedna sekwencja zawiera w całości drugą (z dopuszczonym odsetkiem błędów)

Asemblacja – graf nałożeń

[J.D. Kececioglu i E.W. Myers, *Algorithmica* 13 (1995)] – cd.

S={TATGC,ATCAGCAAC,GCAGCA,GACTC,GTAGA}



Za dopuszczalne nałożenia uznano obejmujące co najmniej dwa znaki każdej sekwencji, przy dwóch znakach bezbłędne, a od trzech w górę zawierające co najwyżej jeden błąd.

Asemblacja – graf nałożeń

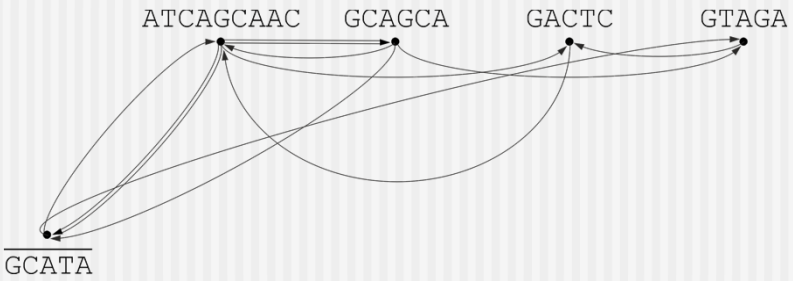
[J.D. Kececioglu i E.W. Myers, *Algorithmica* 13 (1995)] – cd.

- Graf redukowany jest do momentu pozostawienia po jednym wierzchołku z każdej pary i utworzenia struktury lasu skierowanego rozchodzącego się (zbioru prostych ścieżek po pominięciu łuków reprezentujących zawieranie)
- Kwestia wyboru właściwej orientacji ciągów sprowadzona jest do rozwiązania problemu maksymalizacji sumy wag podzbioru łuków w grafie. Na potrzeby przykładu możemy przyjąć, że waga łuku oddaje długość nakładających się fragmentów
- Autorzy zaproponowali algorytm zachłanny, który tworzył podział zbioru wierzchołków, dokładając po jednym maksymalizującym w danej chwili zysk

Asemblacja – graf nałożeń

[J.D. Kececioglu i E.W. Myers, *Algorithmica* 13 (1995)] – cd.

- Graf po pierwszym etapie redukcji

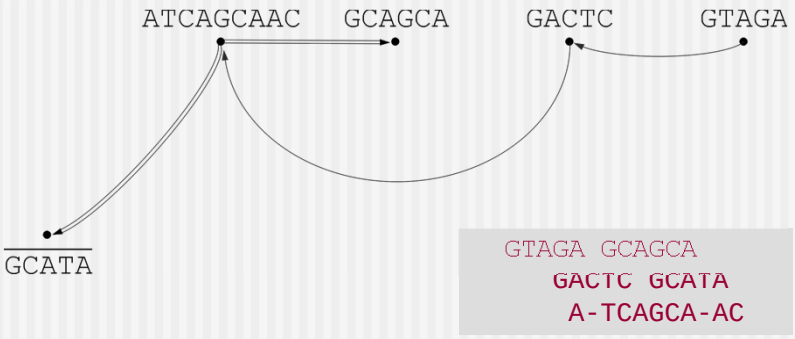


W kolejnym etapie redukowane są nadmiarowe łuki, pozostają te o najwyższej wadze.

Asemblacja – graf nałożeń

[J.D. Kececioglu i E.W. Myers, *Algorithmica* 13 (1995)] – cd.

- Graf po drugim etapie redukcji



Dekompozycja odczytów

- Model obliczeń obecnie najczęściej wykorzystywany w asemblacji odczytów pochodzących z sekwencjonowania nowej generacji opiera się na różnych wariantach grafów z metody Pevznera (błędnie nazywanych *grafami de Bruijna*)
- Grafy powstają przez dekompozycję sekwencji wejściowych (odczytów z sekwenatora) na serie krótszych nakładających się *l*-merów, które reprezentowane są jako łuki. Dopuszcza się tylko bezbłędne nałożenia. Błędna informacja obecna w instancji jest korygowana lub ignorowana
- Rozwiązaniem jest zbiór ścieżek odpowiadających kontigom, włączających w miarę możliwości po jednej sekwencji z pary „czytana wprost – odwrotnie komplementarna”

Asemblacja – graf dekompozycji

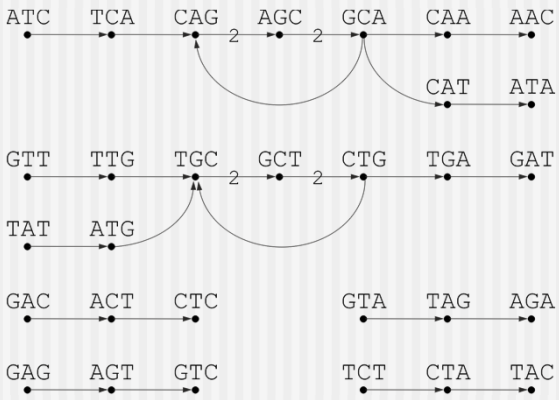
[R. Idury i M. Waterman, *J. Comp. Biol.* 2 (1995)]

- Sekwencje wejściowe i ich odwrotnie komplementarne odpowiedniki dekomponowane są na serie $n–l+1$ l-merów, gdzie n jest długością sekwencji
- Konstrukcja grafu skierowanego z łukami odpowiadającymi l-merom i wierzchołkami odpowiadającymi ich prefiksom/ sufiksom o długości $l–1$. Rozwiązaniem jest zbiór ścieżek pokrywających graf
- W celu zachowania informacji o sekwencjach wejściowych preferowana jest duża wartość l, dodatkowo pamiętana jest przynależność łuków do sekwencji
- Metoda zakłada brak błędów w sekwencjach (insercji, delecji, zamian), ale autorzy dopuścili pominięcie pewnych łuków w rozwiązaniu lub użycie niektórych więcej razy niż przewidziano

Asemblacja – graf dekompozycji

[R. Idury i M. Waterman, *J. Comp. Biol.* 2 (1995)] – cd.

S={TATGC,ATCAGCAAC,GCAGCA,GACTC,GTAGA}, l=4



Liczba wystąpień l-meru nie wskazuje liczby przejść przez dany łuk.

Należy wybrać po jednym reprezentancie z każdej pary zdublowanych kontigów.

Asemblacja – graf dekompozycji

[P.A. Pevzner, H. Tang i M.S. Waterman, *Proc. Natl. Acad. Sci. USA* 98 (2001)]

- Algorytm EULER bazujący na dekompozycji odczytów. Na wejściu sekwencje z błędami pochodzące z obu nici DNA
- Etap korekcji błędów: dążenie do eliminacji maksymalnej liczby potencjalnych błędów w sekwencjach (insercji, delecji, zamian) poprzez relatywnie małą liczbę mutacji. Efektywność mutacji jest mierzona całkowitą liczbą *l*-merów w instancji (włączając w to odwrotnie komplementarne odpowiedniki sekwencji)
- W metodzie mutacji poddawane są *l*-mery o małej liczbie wystąpień i liczba zmutowanych nukleotydów w sekwencji jest ograniczona. Mutacja pociąga za sobą zmiany we wszystkich *l*-merach z danej sekwencji pokrywających ten nukleotyd

Asemblacja – graf dekompozycji

S={TATGC, ATCAGCAAC, GCAGCA, GACTC, GTAGA}

[P.A. Pevzner, H. Tang i M.S. Waterman, *PNAS* 98 (2001)]

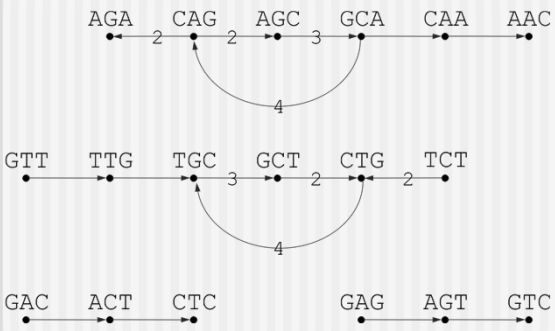
Przykładowa korekcja błędów przy uznaniu *l*-merów występujących jednokrotnie za słabe, dopuszczeniu jednej mutacji na sekwencję i ograniczeniu modelu błędów do tylko zamiany znaków (bez insercji/delecji).

Sekwencja wejściowa	Sekwencja po zmianie	Spektrum po zmianie mocne	Spektrum po zmianie słabe
		CAGC, AGCA, TGCT, GCTG	TATG, ATGC, ATCA, TCAG, GCAA, CAAC, GACT, ACTC, GTAG, TAGA, GCAG, GCAT, CATA, GTTG, TTGC, CTGA, TGAT, GAGT, AGTC, TCTA, CTAC, CTGC
ATCAGCAAC / GTTGCTGAT	AGCAGCAAC / GTTGCTGCT	CAGC, AGCA, TGCT, GCTG, GCAG, CTGC	TATG, ATGC, GCAA, CAAC, GACT, ACTC, GTAG, TAGA, GCAT, CATA, GTTG, TTGC, GAGT, AGTC, TCTA, CTAC
GTAGA / TCTAC	GCAGA / TCTGC	CAGC, AGCA, TGCT, GCTG, GCAG, CTGC	TATG, ATGC, GCAA, CAAC, GACT, ACTC, CAGA, GCAT, CATA, GTTG, TTGC, GAGT, AGTC, TCTG
TATGC / GCATA	TCTGC / GCAGA	CAGC, AGCA, TGCT, GCTG, GCAG, CTGC, CAGA, TCTG	GCAA, CAAC, GACT, ACTC, GTTG, TTGC, GAGT, AGTC

Asemblacja – graf dekompozycji

[P.A. Pevzner, H. Tang i M.S. Waterman, *Proc. Natl. Acad. Sci. USA* 98 (2001)] – cd.

- Graf po korekcji



Po korekcji:

```
AGCAGCAAC
 GCAGCA
 GCAGA
 GCAGA      GACTC
```

Przed korekcją:

```
ATCAGCAAC
 GCAGCA
 GCATA
 GTAGA      GACTC
```

Asemblacja – graf dekompozycji

[D.R. Zerbino i E. Birney, *Genome Res.* 18 (2008)]

- Algorytm Velvet: zastosowanie dekompozycji odczytów i grafu Pevznera. Ścieżki proste (bez rozgałęzień) kumulowane są do jednego wierzchołka, a odcinki odwrotnie komplementarne łączone są w pary
- Etap korekcji błędów: usuwanie krótkich „ślepych” ścieżek, usuwanie ścieżek równoległych do innych o zbliżonej sekwencji, usuwanie ścieżek o zbyt niskim pokryciu
- Rozwiązanie budowane jest z uwzględnieniem oryginalnych odczytów jako wskazówek do przechodzenia przez rozgałęzienia
- W wariancie programu dostosowanym do odczytów sparowanych wierzchołki łączone są mostkami wskazującymi na ich sparowanie. Mostki służą jako wskazówki do budowy rozwiązania

Porównanie podejść

- Mocną stroną modelu opartego na dekompozycji odczytów jest mniejszy wpływ błędów sekwencjonowania na postać generowanego rozwiązania. Tutaj *l*-mer z błędem często nie bierze udziału w tworzeniu rozwiązania, gdyż w danych przeważają *l*-mery bezbłędne powstałe z dekompozycji sąsiednich odczytów — korekcja błędów jest prostsza
- Zastosowana reprezentacja grafu dekompozycji skutkuje mniejszą zajętością pamięci (mniej wierzchołków, mniej łuków), brak błędów nałożeń skraca też czas tworzenia grafu oraz eliminuje etap generowania sekwencji konsensusowej

Porównanie podejść

- Słabą stroną modelu dekompozycji odczytów jest utrata informacji spowodowana rozbiciem ich na krótsze fragmenty. Niektóre algorytmy próbują sobie z tym radzić, odwołując się do odczytów źródłowych w trakcie poszukiwania sekwencji wynikowej. Rośnie jednak wtedy złożoność algorytmu, a braku informacji zwykle nie da się nadrobić w całości (procedura jest heurystyczna)
- Przyjęcie modelu ścieżki Eulera zamiast Hamiltona nie daje w tym problemie zysku na złożoności, obecne dodatkowe ograniczenia czynią problem trudnym obliczeniowo

Graf dla odczytów sparowanych

[P. Medvedev i in., *Lect. Notes Comput. Sci.* 6577 (2011)]

- Zaproponowany został nowy rodzaj grafów (*paired de Bruijn graphs*), w których informacja o sparowaniu odczytów wpływa na ich konstrukcję
- W dotychczas istniejących algorytmach asemblacji informacja o parach jest początkowo ignorowana, tworzone są grafy asemblacji jak dla pojedynczych odczytów i dopiero na etapie budowy ścieżki informacja o parach służy do wskazywania właściwej drogi
- Sposób konstrukcji nowych grafów sprawia, że zwykle są mniej zawikłane, gdyż dopuszcza się połączenia tylko wtedy, gdy nakładają się na siebie oba elementy z dwóch rozważanych par, w zadanej kolejności i w dopuszczalnym przesunięciu

Graf dla odczytów sparowanych

[P. Medvedev i in., *Lect. Notes Comput. Sci.* 6577 (2011)] – cd.

- Zostały wyróżnione dwa typy sparowanych grafów: *idealny* z dokładną odległością pomiędzy odczytami (taką samą dla wszystkich odczytów w instancji) oraz *przybliżony* z odległością z dopuszczonego zakresu $\pm\Delta$
- Podobnie jak w klasycznej metodzie opartej na dekompozycji odczytów, pary odczytów rozbijane są na pary krótszych l-merów związanych z łukami grafu
- Grafy są konstruowane z bezbłędnym nałożeniem l-merów, dla odczytów przetłumaczonych na tę samą nić DNA i o znanej względem siebie orientacji (lewy odczyt w parze jest poprzednikiem prawego). Dla odczytów o nieznanej orientacji proponowane jest dublowanie danych z instancji i budowanie podwójnego grafu

Graf dla odczytów sparowanych

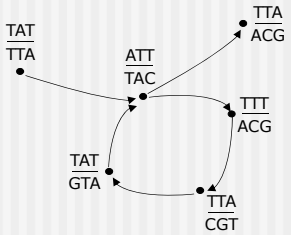
[P. Medvedev i in., *Lect. Notes Comput. Sci.* 6577 (2011)] – cd.

- *W grafie idealnym* dla dokładnych odległości każdą parę odczytów dekomponuje się na serię par *l*-merów. Łuki rozpięte są pomiędzy wierzchołkami reprezentującymi pary ich sufiksów i prefiksów o długości *l*–1
- W *grafie przybliżonym* dodatkowo skleja się wierzchołki, których lewe etykiety pokrywają się, a prawe są względem siebie przesunięte w ramach dopuszczonego zakresu ±Δ. Jeśli $\Delta \geq \frac{1}{2} l$, to zamiast porównywać prawe etykiety dwóch wierzchołków należy obliczyć długość najkrótszej ścieżki w grafie pomiędzy tymi wierzchołkami

Graf dla odczytów sparowanych

- Przykład grafu idealnego z metody Medvedeva i in.

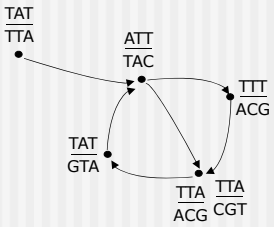
Sparowane odczyty z odległością pomiędzy nimi równą 1:
TATTT+TTACG, ATTTA+TACGT, TTTAT+ACGTA, TTATT+CGTAC, TATTA+GTACG
Sparowane *l*-mery o długości 4:
TATT+TTAC, ATTT+TACG, TTTA+ACGT, TTAT+CGTA, TATT+GTAC, ATTA+TACG



Sekwencja wynikowa:
TATTTATTACGTACG

Graf dla odczytów sparowanych

- Przykład grafu przybliżonego dla progu błędu $\Delta = 1$



Sekwencje wynikowe:
TATTTATTACGTACG
TATTATTTACGTACG

- Dopuszczenie dużej wartości Δ spotykanej w rzeczywistych eksperymentach (nawet do 25% odległości pomiędzy odczytami, odległość nawet do kilku tysięcy nukleotydów) mocno komplikuje postać grafu. Mimo to graf taki będzie źródłem nie większej liczby możliwych ścieżek niż graf tradycyjny