SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

SEMINAR

Faza konsenzusa u OLC paradigmi sastavljanja genoma

Ivan Paljak, Ivan Sekulić
Voditelj: *Mile Šikić*

Zagreb, siječanj 2017.

SADRŽAJ

1. Uvod

Proučavanje strukture DNA u fokusu je znanstvenika od samog otrkića DNA. Precizno čitanje genoma predstavlja velik izazov te su s vremenom razvijani sve brži i jeftiniji uređaji za što bolja očitanja. Danas su uređaji i dovoljno brzi i jeftini, ali se javlja problem kratkih očitanja. Bioinformatika, između ostalog, teži spojiti ta kratka očitanja u jedinstvenu sekvencu (genom).

Prilikom sastavljanja genoma, javlja se nekoliko problema. Prvi je nepreciznost uređaja za sekvenciranje, što zahtjeva višestruka očitanja jednog genoma kako bi se, određenim metodama, mogao ustanoviti stvarni niz. Također, danas se koriste metode temeljene na *shotgun* sekvenciranju cijelog genoma pri čemu nemamo nikakvu informaciju o poretku pojedinih očitanja. Treći otežavajući faktor uspješnog sastavljanja jedinstvene sekvence jest varijabilna duljina očitanja. Uređaji druge generacije, koji treutno prevladavaju, rade očitanja veličine od nekoliko desetaka do par stotina nukleotida. Treća generacija uređaja proizvodi dulja očitanja, od nekoliko tisuća nukleotida, ali imaju velik postotak pogreške – od 15% do čak 40%.

Razvijeno je nekoliko algoritama koji se bave problemom sastavljanja genoma, a najkorišteniji su oni temeljeni na algoritmima nad grafovima. Najčešće se koristi jedna od dviju osnovnih metoda: Preklapanje-Razmještaj-Konsenzus *engl. Overlap-Layout-Consensus, OLC* metode temeljene na grafu preklapanja ili metode temeljene na *de Bruijn* grafovima. U ovom projektu, bavimo se konsenzus fazom OLC paradigme.

Ovaj dokument organiziran je na sljedeći način: u sljedećem poglavlju opisan je algoritam koji smo koristili za dobivanje konsenzusa. Poglavlje 3 kratko opisuje našu konkretnu implementaciju i karakteristike korištenih računala. U poglavlju 4 dana je usporedba vlastite implementacije s referentnim radom. Poslijednje poglavlje sadrži zaključak cjelokupnog projekta.

2. Algoritam Sparc

Cilj ovog projekta bio je implementirati algoritam Sparc (Ye i Ma, 2016) koji se koristi u konsenzus fazi preklapanje-razmještanje-konsenzus (engl. Overlap-Layout-Consensus, OLC) pristupa. Sparc je algoritam za konsenzus fazu OLC pristupa, temeljen na *k-mer/de Bruijn* (Hannenhalli et al., 1996) grafovima.

2.1. Ulaz u algoritam

TODO: seke Algoritam koristi izlaz iz faze razmještanja OLC pristupa te sva početna očitanja genoma. Očitanja su mapirana na kontigu iz faze razmještanja i pohranjena u datoteku formata .sam, opisanom u nastavku.

Kako bi uspješno izgradili graf i proveli algoritam, potrebno je znati gdje se pojedina očitanja mapiraju na osnovnu kontigu. Te informacije dobivano iz .sam datoteke koju smo generirali alatom *GraphMap* (Sovi'c et al., 2016). Nama najvažnije informacije u datoteci jesu:

POS pozicija na osnovnoj kontizi na kojoj počinje mapiranje pojedinog očitanja

CIGAR operacije obavljene nad očitanjem kako bi se dobilo mapiranje (dodavanje, brisanje, pomicanje, itd.)

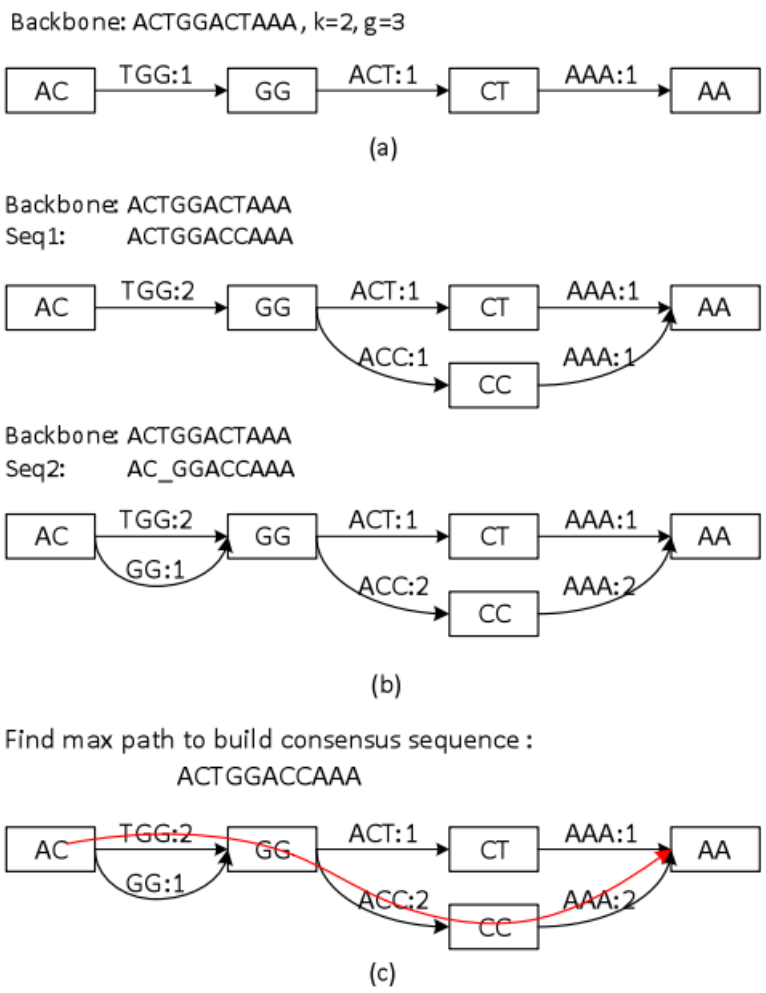
SEQ: očitana sekvenca u prvotnom obliku (bez obavljenih CIGAR operacija)

QUAL kvaliteta očitanja (iz .fastq datoteke očitanja)

Važnost navedenih informacija bit će jasnija kasnije. Za naše potrebe, spremili smo sve važne informacije u zasebnu datoteku koja se sastoji od osnovne kontige (engl. backbone, layout) u prvoj liniji. U nastavku datoteke su očitanja zapisana kroz tri linije. Prva linija predstavlja originalno očitanje na koje su primjenjene CIGAR operacije kako bi se dobilo poravnanje s backbone-om. U drugoj liniji nalazi se kvaliteta očitanja (QUAL), a u trećoj je pozicija na backbone-u na kojoj počinje mapiranje izmjenjene sekvence.

2.2. Opis algoritma

TODO: paljak Najprije se gradi graf direktno iz backbone-a, kako je prikazano na slici **??**. Linearno se prolazi kroz backbone te se u čvorove stavljaju *k-torke (k-meri)*, a bridovi su definirani kao postojeći, ako su sufiks jednog čvora, a prefiks drugoga. Zatim se prolazi kroz mapirana očitanja te se ažuriraju težine u grafu gdje dolazi do preklapanja. Dodaju se novi čvorovi i bridovi ako se u očitanju javljaju nove k-torke. Konačni korak je pronalazak najtežeg puta u grafu. Primjer izgradnje grafa dan je na slici 2.1. TODO: objasni bolje



Slika 2.1: Izgradnja grafa.

3. Implementacija

3.1. Organizacija koda

Cjelokupni algoritam implementiran je u programskom jeziku C++11. Implementacija je podijeljena u dva osnovna dijela. Prvi modul *reading.cpp* učitava sve potrebne podatke (backbone i očitanja u .sam formatu) te primjenjuje CIGAR operacije na pojedina očitanja. Time generiramo datoteku vlastitog formata. U prvoj liniji zapisan je backbone, a svake sljedeće tri linije predstavljaju jedno očitanje na način:

1. očitanje na koje su primijenjene CIGAR operacije,
2. kvaliteta očitanja na koju su primijenjene CIGAR operacije,
3. pozicija u backbone-u na koju se mapira očitanje.

Drugi modul implementira sam algoritam Sparc. TODO: ljakpa

3.2. Kofiguracija korištenog računala

Računalo korišteno pri pokretanju cjelokupnog pipelinea ima sljedeću konfiguraciju:

OS Linux 14.04.1-Ubuntu x86_64

Procesor Intel(R) Core(TM) i7-5820K CPU @ 3.30GHz (CPUs: 12)

RAM 32Gib @ 2133 MHz

4. Evaluacija

Testni podatci dobiveni su na kolegiju[1]. Evaluacija je obavljena na genomima dviju bakterija: escherichia coli te fuge lambda. Za svaku bakteriju dostupna su očitanja, genom nakon faze razmještanja te referentni genom s kojim uspoređujemo performanse svoga algoritma. Cilj je bio generirati genom konsenzus fazom kako bi njegovo preklapanje s referentnim bilo što veće od preklapanja genoma nakon faze razmještanja s referentnim.

4.1. DnaDiff iz MUMmer paketa

MUMmer (Kurtz et al., 2004) je široko korišten paket otvorenog koda (engl. *Open Source*) za razne podzadatke bioinformatike. Implementira razne module, a mi smo koristili *dnadiff* – modul dizajniran za evaluaciju slijedova dvaju vrlo sličnih genoma. Pruža detaljne informacije o razlikama između dvaju genoma, ali generira i high-level datoteku s kvantificiranim razlikama. Svoj algoritam vrednovali smo prema *AvgIdentity* polju u navedenoj datoteci (out.report). Broj predstavlja prosječno poklapanje referentnog genoma s našim. Postoje podatci za 1-na-1 mapiranje (gdje su ponavljanja zanemarena) te M-na-M mapiranje. Uglavnom su oba broja jednaka, pa stoga nije posvećeno previše pažnje na odabir određenoga. U svim rezultatima u nastavku nalazi se *AvgIdentity* polje iz 1-na-1 mapiranja.

4.2. Usporedba vlastitog rješenja s referentnim radom

Tablica 4.1.

[1] fer.unizg.hr/predmet/bio

Tablica 4.1: Usporedba vlastitog rješenja (SpaCRO) s sekvencom iz faze razmještanja (default) i referentnim radom (Sparc).

	lambda	ecoli
default	86.16	88.57
Sparc	**95.41**	**98.17**
SpaCRO	87.93	90.73

Tablica 4.2: Potrošnja memorije i vrijeme izvođenja našeg algoritma na testnim skupovima podataka.

	lambda	ecoli
mem [MB]	30	2464
time [s]	0.45	62.8

4.3. Memorija i vrijeme izvođenja

Kao dio projekta, potrebno je izmjeriti potrošnju memorije i vrijeme izvođenja našega algoritma te zadovoljiti određena ograničenja. Mjerenje je obavljeno pomoću alata cgmemtime[2], a rezultati su prikazani u tablici 4.2. I utrošena memorija i vrijeme izvođenja zadovoljavaju zadana ograničenja.

[2] github.com/isovic/cgmemtime

5. Zaključak

U okviru ovoga projekta upoznali smo se s osnovama bioinformatike. Proučili smo OLC paradigmu sastavljanja genoma te implementirali algoritam temeljen nad grafovima za konsenzus fazu paradigme. Uočili smo određene probleme sastavljanja genoma te se upoznali s postupcima i najčešće korištenim alatima u bioinformatici.

Impementiran je algoritam *Sparc* te je napravljena analiza rezultata. Dobiveni rezultati uspoređeni su referentnim radom. Nismo postigli rezultate kao referentni alg....

6. Literatura

Sridhar Hannenhalli, William Feldman, Herbert F Lewis, Steven S Skiena, i Pavel A Pevzner. Positional sequencing by hybridization. *Computer applications in the biosciences: CABIOS*, 12(1):19–24, 1996.

Stefan Kurtz, Adam Phillippy, Arthur L Delcher, Michael Smoot, Martin Shumway, Corina Antonescu, i Steven L Salzberg. Versatile and open software for comparing large genomes. *Genome biology*, 5(2):1, 2004.

Ivan Sovi'c, Mile Šiki'c, Andreas Wilm, Shannon Nicole Fenlon, Swaine Chen, i Niranjan Nagarajan. Fast and sensitive mapping of nanopore sequencing reads with graphmap. *Nature communications*, 7, 2016.

Chengxi Ye i Zhanshan Sam Ma. Sparc: a sparsity-based consensus algorithm for long erroneous sequencing reads. *PeerJ*, 4:e2016, 2016.

7. Sažetak

Ovaj projekt napravljen je za kolegij Bioinformatika na FER-u. Implementiran je algoritam *Sparc* za generiranje konsenzusa u OLC paradigmi sastavljanja genoma. Dobiveni su rezultati te uspoređeni s referentnim radom.