

# Restrikčné mapy

Michal Čermák & Pavol Gál

# Restrikčné enzýmy

- Strihajú DNA tam kde rozoznajú istú sekvenciu
- Podľa dĺžky trvania experimentu ju rozstrihnú na jednom alebo viacerých rozpoznaných miestach
- Nemusí sa nám podariť získať všetky fragmenty, ktoré chceme

# Čo sú restričné mapy

- Z experimentov poznáme dĺžky fragmentov
- Naša úloha je z ich dĺžok určiť presné polohy všetkých strihov
- Neexistuje polynomiálny algoritmus
- NP problém ??

# PDP vs SPDP

- Obidva exponenciálne
- Vhodne orezávajú vetvy backtrackingu
- V praxi skoro lineárne
- Ktorý je lepší? Rýchlejší, istejší?
- $n$  je počet strihov určených restričným enzýmom

# PDP

- Vykoná sa tzv. čiastočný digest – aby vznikli fragmenty všetkých možných dĺžok, teda fragmenty určené všetkými dvojicami strihov (vieme rozlíšiť rovnako dlhé fragmenty určené rôznymi dvojicami strihov)
- Vstupné dáta rádove  $n^2$
- Zostupne berie dĺžky fragmentov, počíta rozdiely skúmanej dĺžky od ostatných zatiaľ nepoužitých dĺžok a ak sa dá, tak množinu týchto rozdielov vyhodí

# SPDP

- Krátky digest – fragmenty vzniknuté práve jedným strihom (množina gama); jeden strih = 2 fragmenty, spätne nám určujú dve možné umiestnenia strihu
- Úplny digest – fragmenty určené všetkými možnými strihmi (množina lambda)
- Vstupné dáta rádove  $n$
- Menší vstup – bude rýchlejší?

# SPDP

- Berieme postupne všetky usporiadané dvojice doplňujúcich sa fragmentov z gammy (každá dvojica dvakrát)
- Postupne vytvárame podmnožinu možných pozícií rezov spolu s množinou fragmentov aké ju určujú
- Táto množina fragmentov je porovnávaná s lambdou (pri rovnosti máme riešenie)

# Dátové štruktúry

- V oboch algoritmoch sa používajú multimnožiny a prevádzajú sa s nimi množinové operácie
- My si ich udržiavame pokiaľ možno usporiadané a všetky množinové operácie potom uskutočňujeme v lineárnom čase



# Dajú sa porovnať?

- Väčšina restriktčných enzýmov strihá genóm na veľmi málo miestach (< 20)
- Oba algoritmy skončia prakticky hneď pre takéto malé vstupy
- AluI je jediný enzým, ktorý strihá genóm na niekoľko desiatok miestach (match na AGCT) a práve pri ňom začína SPDP výrazne zaostávať v rýchlosti

# Spočítame rekurzívne zavolania

- Nemá zmysel merať reálny čas behu, pretože nás zaujíma práve ako úspešne orezávajú tieto algoritmy vetvy backtrackingu
- Pre ich porovnanie spočítame počet rekurzívnych zavolaní algoritmu
- Pri veľkých vstupoch (AluI enzým) sa PDP zavolá len rádove niekoľko 100 krát, kým SPDP neuveriteľných niekoľko 100 000 až miliónov krát!

# PDP vs SPDP

- Ukázalo sa, že PDP orezáva vetvy backtrackingu ďaleko lepšie ako SPDP
- V praxi sa napriek tomu používa SPDP
- Experimenty na získanie vstupu pre SPDP sú totiž spoľahlivejšie
- V krátkom digeste ľahko overíme, či nám chýba doplnkový fragment
- Úplny digest stačí nechať bežať dostatočne dlho
- V čiastočnom digeste je ťažké získať VŠETKY fragmenty!!!