ČLANEK 1
VERZIJA 1

DNK računanje

Mojca Pregelj

1. Uvod

1.1 *Splošen pregled DNK računanja*

Ene izmed najbolj pomembnih molekul, ki se nahajajo v celicah vsakega živega bitja so **nukleinske kisline**. Grobo lahko razdelimo nukleinske kisline v **deoksiribonukleinske kisline** (DNK) in **ribonukleinske kisline** (RNK). DNK je organska snov, ki jo sestavljajo organska baza, sladkor in fosforjeva kislina ref[1]. Poleg tega, da ta molekula nosi genski zapis informacij pomembnih za obstoj in življenje celice, jo lahko uporabimo tudi za izvajanje računskih operacij in shranjevanje poljubnih informacij. Računanje z uporabo DNK se imenuje **DNK računanje** ref[25]. Prednost DNK računanja pred običajnim računanjem najdemo v lastnostih molekule deoksiribonukleinske kisline, ki omogočajo izvajanje več operacij istočasno na tej molekuli. Zaradi tega DNK računanje ponuja občutno hitrejšo povprečno hitrost reševanja zapletenih problemov v primerjavi z običajnim računalnikom in shranjevanje velike količine informacij na majhni prostornini. Izračunali so, da za razbitje najbolj znanega **simetričnega tajnopisa** (DES-a) običajen računalnik potrebuje 10000 let, da preveri vseh 2^{56} možnih ključev ref[25]. Z DNK paralelnim računanjem pa bi se lahko napad na DES izvršil v 6715 korakih, kar bi ob predpostavki, da trajanje vsakega koraka reduciramo le na 1 sekundo, pomenilo, da lahko DES razbijemo celo le v 2 urah ref[3].

1.2 *Kratka zgodovina DNK računanja*

DNK računanje se je kot koncept rodilo leta 1993, ko je Leonard Adleman, matematik specializiran za računalniško znanost in kriptografijo, med branjem knjige J. Watsona, slučajno naletel na podobnosti med običajnim računalnikom in molekulo DNK. Po malo več kot enem letu je Adleman uporabil svoj DNK računalnik za rešitev problema Hamiltonske poti in s tem je bila DNK prvič uporabljena za računanje. Adlemanovo objavljeno delo je bilo sprejeto z mešanimi občutki - nekateri so v tem delu videli le dober izračun, drugi pa novo generacijo računalnikov. V decembru istega leta je Richard Lipton napisal članek, v katerem je opisal metodo, po kateri lahko zaporedje baznih parov DNK spremenimo v dvojiško zaporedje. S tem je lahko DNK računalnik uporabljal isto osnovno strukturo (Boolovo logiko) kot moderni računalniki. To je pomenilo, da je DNK računalnik v teoriji postal enako programabilen kot običajni računalniki ref[25].

Preostanek članka je organiziran takole:
V 2. poglavju je obrazložena zgradba deoksiribonukleinske kisline in tvorba dvojne vijačnice.
V 3. poglavju so našteti najpomembnejši osnovni mehanizmi DNK računanja, ki se pojavljajo v članku. 4. poglavje predstavi prednosti in uporabo DNK računanja. Najprej razkriva možnosti uporabe molekule DNK v kriptografiji. Pri tem je najprej predstavljena genska koda, nato pa sta v nadaljevanju poglavja obravnavana dva različna DNK kriptosistema. To poglavje prikaže še

zanimiv požrešni napad na DES s pomočjo DNK računanja. Zaključi se s problemom Hamiltonske poti ter DNK predstavi kot Turingov stroj. V 5. poglavju bralec spozna pomanjkljivosti in ovire DNK računanja, torej se tukaj srečamo z napakami in hitrostjo pri takšnem računanja. Seminar se konča s 6. poglavjem, to je pogledom v prihodnost, kjer lahko svoji domišljiji pustimo prosto pot.

2. Biološka zgradba DNK

V tem poglavju je obrazložena zgradba deoksiribonukleinske kisline in nato opisana tvorba dvojne vijačnice.

2.1 *Zgradba deoksiribonukleinske kisline*

Deoksiribonukleinska kislina je organska snov, ki jo sestavljajo **organska baza**, **sladkor** (deoksiriboza) in **fosforjeva kislina** ref[1].
Nukleinske kisline so visokomolekularne organske spojine, katerih molekulska masa varira od nekaj milijonov do nekaj milijard, molekule pa so dokaj stabilne in toge. Sestavljene so iz manjših enot, ki jih imenujemo **nukleotidi**. Vsak nukleotid pa je nadalje sestavljen iz treh različnih molekul: **organske baze**, **sladkorja pentoze** (deoksiriboza) in **ostanka fosforjeve kisline** ref[10].
Slika 1.

Slika 1 Kratek odsek nukleinske kisline. S črticami omejeno polje označuje enoto nukleinske kisline – nukleotid ref[10].

Deoksiribonukleinska kislina je v bistvu zelo dolga nerazvejana spojina, sestavljena iz **velikega števila medsebojno vezanih molekul** (monomerov) - nukleotidov, ki se lahko vežejo med seboj v dolgo verigo (polimer).

Organske baze, ki so sestavni del nukleotidov, so dobile ime po tem, ker vsebujejo bazične – NH_2 skupine. Po kemični sestavi ločimo baze z enim obročem in baze z dvojnim obročem. Baze z enim obročem so: **citozin** (C), **timin** (T) in **uracil** (U), bazi z dvojnim obročem pa sta **adenin** (A) in **guanin** (G). Glede na to, katera od navedenih baz sestavlja nukleotid, razlikujemo pet vrst nukleotidov: adenin-nukleotid, gvanin-nukleotid, citozin-nukleotid, timin-nukleotid in uracil-nukleotid ref[13].

Fosforjeva kislina daje nukleinskim kislinam kislo naravo ter povezuje nukleotide v verigo. Veriga nastane tako, da se fosforna kislina enega nukleotida poveže s sladkorjem sosednjega nukleotida in tako nastane močna kemijska vez ref[10].

2.2 *Tvorba dvojne vijačnice*

Analize molekule DNK so pokazale, da je število adenin-nukleotidov enako številu timin-nukleotidov, prav tako pa tudi število gvanin-nukleotidov povsem ustreza številu citozin-nukleotidov. To lahko izrazimo z enačbama: #A = #T ter #G = #C.

Raziskave so pokazale, da molekulo DNK sestavljata dve verigi, zviti v obliki vijaka. Na osnovi teh podatkov sta Watson in Crick izdelala model molekule DNK in ga objavila leta 1953. Obe verigi dvojne vijačnice se med seboj povezujeta s šibkimi vodikovimi vezmi med organskimi bazami posameznih nukleotidov obeh vijačnic. Posebnost v tej zgradbi je način, kako sta verigi povezani med seboj. Kemična narava organskih baz terja, da se adenin lahko povezuje samo s timinom in citozin le z gvaninom. Takšno povezovanje, ki mu pravimo tudi **parjenje baz**, zagotavlja, da sta si vijačnici komplementarni ref[10]. Slika 2.

Slika 2. Skica dvojne vijačnice. Verigi iz sladkorjev in fosfatov se zvijeta druga okoli druge na zunanji strani molekule, notranjost pa zapolnjujejo pari baz, ki se vežejo z vodikovimi vezmi druga na drugo ref[4,10].

3. Osnovni mehanizmi DNK računanja

V tem poglavju so našteti osnovni biokemični mehanizmi, ki omogočajo DNK računanje.

Rekombinantna DNK – veriga DNK, ki se tvori z novo kombinacijo genetskega materiala.

1. Postopka za tvorbo oziroma razgradnjo posamezne enojne verige sta:
   - **Sintetiziranje** (synthesizing) – konstrukcija enojne verige DNK določene dolžine z dodajanjem nukleotida za nukleotidom;
   - **Uničevanje** (destroying) – označevanje verig z encimi ter odstranitev preostalih verig z gelsko elekroforezo (ločevanje nukleinskih kislin po velikosti s pomočjo gela in električnega toka).

2. Postopka, ki se nanašata na povezovanje oziroma ločevanje posameznih verig v dvojne verige DNK, sta:
   - **Tvorba baznih parov** - hibridizacija (annealing, base paring, hybridization) – povezovanje enojnojnih komplementarnih verig DNK v dvojno verigo DNK;
   - **Denaturacija** (melting, denaturation) – segrevanje dvojne vijačnice DNK, da razpade v enojni vijačnici.

3. Postopka za pomnoževanje in razvrščanje verig DNK sta:

- **Pomnoževanje** (amplifying, copying) – uporaba metode pomnoževanja enojnih verig DNK za povečanja števila DNK verig (polimerase chain reaction – PCR, denaturacija – razdvajanje dvojne verige DNK, tvorba baznih parov enojne vijačnice DNK s startnim zaporedjem nukleotidov, podaljševanje začetnega zaporedja z encimom DNK polimerazo);
- **Ločevanje** (separating) – razvrščanje verig DNK po dolžini s pomočjo električnega polja v gelu (elektroforeza na gelu).

4. Postopka, ki se nanašata na vezi znotraj posamezne verige DNK, sta:
   - **Rezanje** (cutting) – prekinitev verige DNK z uporabo restrikcijskih encimov na določenem mestu;
   - **Povezovanje** (ligating) – spajanje dveh ali več verig DNK v enojno verigo s pomočjo encimov ligaz.

5. Še dva postopka:
   - **Substitucija** (substituting) – zamenjava, vstavljanje ali izbrisanje določenega zaporedja (varianta PCR);
   - **Sekvencioniranje** (sequencing) – postopek branja rezultatov, določanje zaporedja nukleotidov v verigi (kombinacija PCR metode in gelske elektroforeze).

4. DNK kriptografija

V tem poglavju bom prikazala nekaj možnosti, ko lahko molekulo DNK uporabimo tudi v kriptografiji. Najprej je podana kratka ponovitev osnov kriptografije s poudarkom na Vernamovem enkratnem ščitu, nato pa se bralec seznani z načinom kodiranja aminokislin z zaporedjem nukleotidov. V nadaljevanju poglavja sta podrobno obravnavana dva različna DNK kriptosistema. Poglavje se zaključi z opisom postopka molekularnega računanja za napad na enega izmed najbolj razširjenih in uporabljenih kriptosistemov, to je na DES.

Osnove kriptografije

Definicija 1
Kriptosistem je peterica $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ za katero velja:
1. $\mathcal{P}$ je končna množica možnih čistopisov,
2. $\mathcal{C}$ je končna množica možnih tajnopisov,
3. $\mathcal{K}$ je končna množica možnih ključev.
4. Za vsak ključ $K \in \mathcal{K}$ imamo šifrirni postopek $e_K \in \mathcal{E}$ in ustrezen dešifrirni postopek $d_K \in \mathcal{D}$. $e_K: \mathcal{P} \to \mathcal{C}$ in $d_K: \mathcal{C} \to \mathcal{P}$ sta taki funkciji, da je $d_K(e_K(x)) = x$ za vsak $x \in \mathcal{P}$.

Definicija 2
Kriptosistem (P, K, C) ima **popolno varnost**, če je $p_P(x/y) = p_P(x)$ za vse $x \in P$ in $y \in C$, tj. končna verjetnost, da smo začeli s tajnopisom x pri danem čistopisu y, je identična z začetno verjetnostjo čistopisa x.

Izrek 1
Če za kriptosistem $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ velja $|\mathcal{K}| = |\mathcal{C}| = |\mathcal{P}|$, potem ima ta kriptosistem popolno varnost, če in samo če je vsak ključ uporabljen z enako verjetnostjo $1/|\mathcal{K}|$ ter za vsak čistopis x in vsak tajnopis y obstaja tak ključ K, da je $e_K(x)=y$.

Najbolj znana realizacija popolne varnosti je Vernamov enkratni ščit, ki ga je leta 1917 patentiral Gilbert Vernam za avtomatizirano šifriranje in dešifriranje telegrafskih sporočil.

Vernamov enkratni ščit:
Naj bo $\mathcal{P}= \mathcal{C}=\mathcal{K}= (Z_2)^n$, $n\in N$, $e_K(x) = x$ XOR K, deširfiranje je identično šifriranju.

Shannon je bil prvi, ki je po 30 letih dokazal, da tega sistema res ne moremo razbiti. Slaba stran tega kriptosistema je $|\mathcal{K}| \geq |\mathcal{P}|$ in dejstvo, da moramo po vsaki uporabi zamenjati ključ ref[26].

XOR operacija (oznaka je $\oplus$):
$0 \oplus 0 = 0, 0 \oplus 1 = 1, 1 \oplus 0 = 1, 1 \oplus 1 = 1.$

4.1 *Genska koda*

Pravila, po katerih se zaporedje nukleotidov prevede (kodira) v zaporedje aminokislin v beljakovinah, se imenuje **genska koda**.
Glede na to, da so beljakovine sestavljene iz 20 različnih vrst aminokislin, se zastavi vprašanje, kako lahko 4 različni nukleotidi določajo vsako posamezno aminokislino. Kombinacija, s katero lahko označimo vsako posamezno vrsto aminokislin, mora vsebovati 3 nukleotide (**triplet**), saj na tak način dobimo $4^3 = 64$ možnosti. To pa je dovolj in celo preveč, da zaznamujemo vseh 20 vrst aminokislin. Enota treh nukleotidov, ki določajo posamezno aminokislino, se imenuje **kodon**. Tako je večina aminokislin določena z več različnimi kodoni. Trije kodonski tripleti od 64 možnih ne nosijo zapisa za aminokisline temveč služijo kot signal za zaustavitev prevajanja - poimenovani so **stop kodoni**. Slika 6.

Slika 6. Genska koda. Vsaka aminokislina je v nukleinskih kislinah kodirana s tremi nukleotidi. V zgornji tabeli s kombinacijo treh nukleotidov, prvega navedenega na levi strani, drugega nukleotida navedenega zgoraj in tretjega nukleotida navedenega na desni strani, dobimo zaporedje treh nukleotidov, ki kodirajo aminokislino, ki je na presečišču stolpca oziroma vrstic določenih z izbranimi nukleotidi. Imena aminokislin so okrajšana s tremi črkami ref[4].

Odvisno od tega, na katerem od treh nukleotidov v tripletu se začne proces dekodiranja, ločimo tri **bralne okvirje**. Bralni okvir je določen v začetku prevajanja in je nato nespremenjen do konca. Slika 7.

Slika 7 Trije različni bralni okvirji. V prvem primeru se je dekodiranje pričelo s prvim nukleotidom v zaporedju nukleotidov (C), tako tvorjeni prvi kodonski triplet sestavljajo nukleotidi CUC, kar se prevede v aminokislino Leu (Glej sliko 5). V drugem primeru se je dekodiranje pričelo z drugim nukleotidom v zaporedju (U) in tako se je tvoril prvi triplet UCA, ki ustrezna nastali aminokislini Ser. V tretjem primeru pa se je dekodiranje pričelo s tretjim nukleotidom v verigi (C) in tako je prvi triplet sestavljen iz nukleotidov CAG, kar ustreza aminokislini Gin. Podobno, kot se določi prva aminokislina, se določijo še vse naslednje v zaporedju. Glede na to, kje se je začelo prevajanje, se tvorijo tri različna zaporedja aminokislin (Na sliki označena z zeleno barvo) ref[4].

4.2 *DNK kriptosistemi na podlagi naključnih enkratnih ščitov*

Deoksiribonukleinsko kislino lahko uporabimo tudi v kriptografiji in tvorimo kriptografske sisteme na osnovi te molekule. Takšno kriptografijo imenujemo **DNK kriptografija**. Ena izmed prednosti uporabe molekule DNK je v tem, da DNK zagotavlja mnogo bolj kompakten shrambeni medij kot so običajni shrambeni mediji, saj že izjemno majhne količin te molekule zadoščajo za velike kriptografske sisteme.

Vhodni podatki za DNK kriptosistem na podlagi naključnih enkratnih ščitov so kratki odseki različnih čistopisov. Enkripcija se izvede s pomočjo **naključnega šifranta**, ki spremeni kratka zaporedja čistopisa v zaporedja tajnopisa. Naključni šifrant je sestavljen iz niza ključev, ki so naključni in se nikoli ne ponovijo. Za zagotavljanje tajnosti šifriranja sta torej ključni dve točki: popolna naključnost in le enkratna uporaba posameznega ključa.

Za tvorbo naključnega šifranta v obliki DNK je najprej potrebno tvoriti dolga zaporedja nukleotidov v obliki DNK verige, ki je bila naključno sestavljena iz kratkih nukleotidnih zaporedij. Tako tvorjeno verigo je nato potrebno ločiti (separating), s čimer se odstranijo krajša zaporedja nukleotidov, ki se niso pravilno vezala v verigo. Ločevanju sledi pomnoževanje (amplifying, kloniranje) take verige, s čimer se število verig pomnoži in jih je mogoče razdeliti med pošiljalca in prejemnika. Vzemimo, da je bila ta veriga DNK sestavljena tajno. Nadalje predpostavimo, da je ta veriga, tvorjena po principu enkratnega ščita, vnaprej deljena med pošiljalcem in prejemnikom tajnega sporočila. Ta predpostavka zahteva začetno komunikacijo med pošiljateljem in prejemnikom, ki je olajšana z veliko naravno zgoščenostjo in obstojnostjo DNK v raztopini.

Obstaja nekaj postopkov, ki jih uporablja DNK kriptografija na podlagi naključnih enkratnih ščitov, ki imajo to lastnost, da jih ni mogoče razbiti. Podrobno si bomo pogledali dva DNK kriptosistema na podlagi naključnih enkratnih ščitov:

1. **substitucijski** DNK kriptosistem na podlagi enkratnih ščitov ter
2. **DNK XOR** kriptosistem na podlagi enkratnih ščitov.

➢ Substitucijski DNK kriptosistem na podlagi enkratnih ščitov

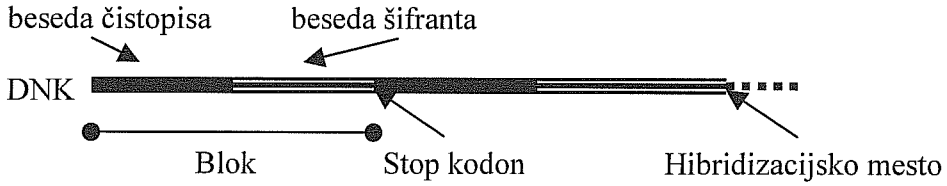
Vhod za substitucijski DNK kriptosistem na podlagi enkratnih ščitov je sestavljen iz:

- čistopisa v obliki dvojiškega sporočila dolžine n, ki je razdeljen v besede določene dolžine in

- naključnega šifranta, ki je sestavljen iz **pregledne tabele**, ki naključno povezuje vse možne verige besed čistopisa v šifrirane besede določene dolžine tako, da obstaja enolično obrnljivo pretvarjanje.

Enkripcija čistopisa s substitucijo se izvede tako, da se vsaka beseda čistopisa spremeni v besedo tajnopisa. Sprememba besede čistopisa skupaj s substitucijskim zaporedjem enkratnega ščita je podana s tabelo. Dekripcija pa se izvede tako, da se izvršijo obratni substitucijski postopki.

V primeru enkripcije s substitucijo želimo spremeniti eno epruveto kratkih DNK enojnih verig (besede čistopisa) v drugo množico popolnoma različnih enojnih verig (besede tajnopisa) na naključen, a vendar obrnljiv način. Pri tem se čistopis spremeni v šifrirano verigo, verigo čistopisa pa nato odstranimo. Celotna shema je sestavljena iz dolgih DNK verig, ki vsebujejo mnogo **blokov**. Vsak blok vsebuje besedo šifranta, ki ji sledi beseda iz čistopisa. Slika 9.



Slika 9 DNK izvedba. Beseda iz šifranta deluje kot **hibridizacijsko mesto** (mesto na DNK verigi, ki ima nukleotide, ki se zaradi svoje sestave vežejo z nukleotidi že pripravljene krajše verige DNK, ki se prilega osnovni verigi) za vezavo krajše veride DNK, ki se po vezavi podaljša z dodajanjem nukleotidov ob besedo iz čistopisa in tako se tvori besedni par. Dodajanje nukleotidov se konča pri stop kodonu. Te pare DNK verig uporabimo kot pregledno tabelo v začetnem koraku pretvorbe čistopisa v tajnopis.

➢ DNK XOR kriptosistem na podlagi enkratnih ščitov

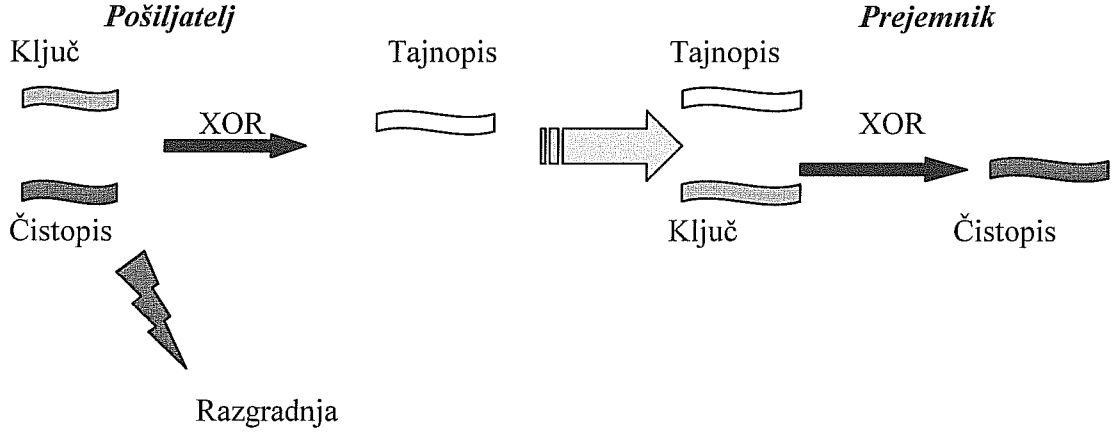
DNK XOR kriptosistem na podlagi enkratnih ščitov temelji na tvorbi zaporedja S iz R neodvisno razporejenih naključnih bitov Pri tem se DNK enkratni ščiti verige pomnožijo in ena kopija verige se shrani pri izvoru, druga kopija pa pri prejemniku sporočila.

Naj bo L število bitov zaporedja S, ki ostanejo neuporabljeni, če je v začetku $L = R$. Ko želimo poslati vhoden dvojiški čistopis P_i, ki je dolg $n < L$ bitov, na vsakem bitu P_i izvedemo XOR operacijo z bitom $K_i = SR - L + i$ in dobimo enkripcijski bit $C_i = P_i$ XOR K_i za i=1, ..., n. Porabljene n bite zaporedja S uničimo pri pošiljatelju, zaporedje tajnopisa C=(C_1, C_2, ..., C_n) pa odpošljemo prejemniku. Pri prejemniku se ponovi enak postopek, le da se zaporedje C uporabi namesto P, uporabi se obratna operacija XOR z biti iz zaporedja S. Po uporabi se biti iz zaporedja S uničijo. Komutativna lastnost operacije XOR omogoča reprodukcijo začetne informacije, saj velja C_i XOR $K_i = P_i$, ker je $C_i = P_i$ XOR K_i in P_i XOR K_i XOR $K_i = P_i$.

DNK kodiranje omogoča te operacije nad modularno osnovo 4, torej v štiriškem sistemu in ne le v dvojiškem, saj je DNK sestavljena iz štirih različnih nukleotidov. V tem primeru lahko za vhodni čistopis in enkratni ščit zaporedje uporabimo zapis v štiriškem sistemu in za kodiranje

uporabimo operacije štiriškega sistema namesto operacije XOR. Vendar si delo poenostavimo in vzamemo, da je informacija v DNK shranjena le v obliki niza bitov, torej v dvojiškem sistemu.

Tako kot pri substitucijskem DNK kriptosistemu na podlagi enkratnih ščitov želimo tudi v tem primeru pretvoriti epruveto kratkih DNK verig (čistopis) v nov niz popolnoma drugačnih verig (tajnopis) na naključen, a vendar obrnljiv način. Z znanimi osnovnimi mehanizmi DNK računanja (npr. povezovanje verig) ref[4] in po metodi enkratnega ščita povežemo **zaporedja čistopisa** z **zaporedji ključa** v posamezne pare. Tako dobljene pare lahko naprej povežemo v novo daljše enotno zaporedje. Deli čistopisa skupaj z zaporedji ključa so z operacijo XOR pretvorjeni v **zaporedja tajnopisa**. Po končani operaciji so zaporedja čistopisa odstranjena. Obratno dekodiranje je podobno, saj uporablja komutativne lastnosti XOR operacije. Slika 11.



Slika 11 Shematski prikaz izvedbe šifriranja z uporabo tehnike XOR in DNK verig. Pošiljatelj z uporabo tehnike XOR tvori iz DNK verige ključa in DNK verige čistopisa verigo DNA tajnopisa. Po tvorbi tajnopisa pošiljatelj razgradi verigo čistopisa. Prejemnik pa zopet z uporabo tehnike XOR iz DNK verige tajnopisa in DNK verige ključa tvori besedo čistopisa.

4.3 *Napad na DES s pomočjo DNK računanja*

Avtorji članka ref[3] so razvili poseben postopek molekularnega računanja za napad na enega izmed najbolj razširjenih in uporabljenih kriptosistemov, to je na DES. Raziskovali so tako imenovan čistopis-tajnopis napad, pri katerem poznamo čistopis ter ustrezen tajnopis, določiti pa želimo ključ, ki je izvedel enkripcijo. Za določitev tega ključa so uporabili metodo požrešnega napada, pri kateri preverimo vseh 2^{56} ključev tako, da izvedemo enkripcijo pod kontrolo vsakega ključa dokler ne najdemo ključa, ki proizvede tajnopis.

Ta postopek je zaenkrat še popolnoma teoretične narave in ni še jasno, če ga je mogoče izvesti v laboratoriju. V tem postopku je uporabljen poseben model molekularnega računanja, ki so ga imenovali »sticker« model. Pri tem potrebujemo napravo (»**sticker**« stroj), ki bi bila primerne velikosti, da bi jo lahko položili na mizo. Prednost tega postopka je v tem,da bi potrebovali le

približno gram DNK molekule, napad bi se izvršil v relativno kratkem času in lahko uspel tudi ob prisotnosti velikega števila napak. ref[3].

Ob danem paru čistopis-tajnopis se algoritem postopka izvede v treh korakih:

1. Vhodni korak: spominske verige DNK molekule predstavljajo vseh 2^{56} ključev.
2. Enkripcijski korak: na vsaki spominski verigi izračuna tajnopis glede na enkripcijo čistopisa pod določenim ključem.
3. Izhodni korak: izbere se tista spominska veriga, katere tajnopis se ujema z danim tajnopisom; prebere se ustrezen ključ.

Večina dela se izvede v drugem koraku, kjer se pojavi DES enkripcija.

Opisani algoritem se izvede na »sticker« stroju, ki ga avtorji ref[23] imenujejo »paralelna robotska delovna postaja«. Stroj je sestavljen iz stojal za epruvete, nekaj robotike (roke, črpalke, grelci, spojniki) in mikroprocesorja, ki nadzira delovanje robotike.

Za to, da najdemo ključ za DES-ov par čistopis-tajnopis, moramo izvesti skupno 6715 korakov. Dejanski čas, ki je potreben za izvedbo algoritma, je odvisen od tega, kako hitro lahko potekajo operacije. Če bi vsaka operacija potekala 1 dan, bi za celoten izračun potrebovali 18 let; če bi vsaka operacija potekala 1 uro, bi za izračun potrebovali približno 9 mesecev; če bi lahko vsako opercijo izvedli v 1 minuti, bi računanje vzelo 5 dni. Če pa bi efektivno trajanje koraka reducirali na 1 sekundo, bi skupno potrebovali le 2 uri ref[3].

»Sticker« stroj naj bi vseboval 1303 epruvet z raztopinami DNK molekul. Ker 2^{56} spominskih verig tehta približno 0.7 g in v raztopini 5 g/l zavzame približno 140 ml, torej ni potrebno, da bi bila prostornina 1303 epruvet večja od 140 ml za vsako epruveto. Sledi, da stojalo z epruvetami zavzema največ 182 l. Tako je lahko naprava 1m široka in dolga ter 18 cm globoka. Prostornina, ki jo zavzema robotika, znaša približno 13 l, to je 1/14 celotne prostornine naprave. Zato sklepamo, da bi lahko celotno napravo postavili na mizo ref[3].

S tem so znanstveniki pokazali, da je možno »realne probleme« rešiti z majhnimi napravami, ki ne potrebujejo velikih količin DNK.

5. Pomanjkljivosti in ovire

V tem poglavju se bralec seznani z nekaterimi pomanjkljivostmi in ovirami DNK računanja. To so predvsem različne vrste napak ter ovirami pri hitrosti DNK računanja.

5.1 *Napake*

Največji problem DNK računanja so napake ref[9]. Obstaja veliko virov napak, vendar pa največ napak povezanih z DNK računanjem izvira iz uporabe računskih tehnik. Verige DNK se lahko med prenosom fizično izgubijo, DNK se lahko poškoduje zaradi prekomerne uporabe ali predolgega čakanja v raztopini, DNK se tudi vedno ne obnaša tako, kot bi želeli da se. Izvaja

lahko kakšne zavoje, pa tudi povezave med nukleotidi niso vedno ustrezne. Znanstveniki se zavedajo problema napak in zato razvijajo številne obetajoče proti ukrepe: načrtno kodiranje DNK, izvajanje operacij s pomočjo drugih tehnik, proučevanje drugih optimalnih reakcijskih okoliščin, uporaba odvečnih elementov in ponavljajočih se poskusov za povečanje zanesljivosti, uporaba drugih molekul kot prenašalcev informacij (sintetične variante DNK). Eno izmed možnih rešitev teh problemov pa je tudi v tem, da se DNK uporablja v povezavi z peptidno hrbtenico in tako poveča stabilnost molekule.

5.2 *Hitrost*

Zaradi sposobnosti DNK molekule, ki lahko izvaja več operacij istočasno, DNK računanje ponuja občutno hitrejšo povprečno hitrost reševanja zapletenih problemov v primerjavi z običajnim računalnikom ref[25]. Sama zase potrebuje DNK molekula približno 1000 sekund, da izvede eno operacijo, oziroma 0,001 milijona navodil na sekundo. Ker lahko najhitrejši superračunalnik izvede 1000 MIPS skozi serijsko logiko, je DNK računanje neuspešno pri izvajanju preprostih operacij. Ker pa DNK računanje poteka paralelno, se lahko izvede veliko operacij istočasno in zato lahko stopnja doseže tudi 10^{14} MIPS. Za razbitje DES-a bi na primer povprečen računalnik potreboval 10000 let, da bi preveril vseh 2^{56} možnih ključev, z DNK paralelnim računanjem pa bi se lahko to izvedlo v 6715 korakih , za katere bi potrebovali približno 5 dni.

6. Pogled v prihodnost

Eno izmed pričakovanj DNK računanja je ideja, da bi današnje računalnike nekoč nadomestili DNK računalniki.

6.1 *DNK računalniki*

Projekt o DNK računalniku so začeli razvijati na univerzi v Wisconsinu, v ZDA. Prednost DNK pristopa je v tem, da DNK računalnik dela paralelno, torej procesira vse možne odgovore istočasno. Elektronski računalnik namreč lahko analizira le en potencialni odgovor naenkrat. Torej bi bil DNK računalnik za večje probleme verjetno hitrejši od običajnega računalnika, še posebej, če bi se izvedle dodatne izboljšave pri sintezi DNK in bi dekodiranje DNK potekalo hitreje.

V članku ref[15] so pri svojem pristopu uporabili pozlačen kvadratek stekla, nekaj podobnega kot običajen spominski čip, ki je predstavljal ključ za izgradnjo DNK računalnika. Na ta pozlačen kvadratek stekla naj bi se zasidralo do trilijona individualnih verig DNK in vsaka veriga naj bi vsebovala informacije, ki bi bile shranjene v DNK računalniku. Vsaka DNK veriga bi predstavlja en možen odgovor za problem, ki ga skuša računalnik rešiti. Tako bi se proizvedla skupina DNK verig, ki bi vsebovale vse možne odgovore. Vendar pa se mora za vsak nov problem narediti

nova množica verig. Večina možnih odgovorov je napačnih, le eden ali nekaj je pravilnih, zato je naloga računalnika, da pregleda vse rešitve in loči pravilne od nepravilnih. To naredi DNK računalnik tako, da vse verige istočasno podvrže seriji kemičnih reakcij, ki posnemajo matematične operacije, ki bi jih izvajal elektronski računalnik.

Zagovorniki DNK računalnika trdijo, da bi tehnologija lahko proizvedla DNK računalnike, ki bi bili pri reševanju določenih problemov boljši celo od sedanjih superračunalnikov. Toda ta vizija je še daleč, saj je sedanja tehnologija DNK računanja tako osnovna, da so poskusi omejeni na vprašanja, za katere človek sploh ne potrebuje računalnika.

6.2 *Pričakovanja DNK računanja*

Največji problem pri DNK računanju je najverjetneje v tem, da so naša pričakovanja prevelika, saj mora DNK računanje preiti še mnogo ovir preden bo univerzalno molekularno računanje postalo resničnost. Z možnostjo avtomatizacije različnih restrikcijskih encimskih reakcij, razvrščanjem, ločevanjem DNK procesov ter PCR ojačevalnim postopkom, bi se namreč stopnja DNK računanja lahko še večala. Vendar pa je DNK računanje precej nova ideja in ima pred seboj še dolgo pot do tega, da postane zrela tehnologija. Na tej poti je potrebno proučiti še nekaj parametrov, kot so velikost eksperimentov, hitrost in zanesljivost posamezne operacije, zanesljivost informacijskega prenašalca ter število zaporednih operacij v eksperimentu. Nemogoče je napovedati, kam bo pot peljala. Četudi je malo verjetno, da bi DNK računanje v bližnji prihodnosti v celoti nadomestilo standardno računanje in bi se izkazalo, da pot ne bo vodila nikamor, se je splačalo potovati po njej.

7. Literatura

1. Slovar slovenskega knjižnega jezika. 1 edition. Ljubljana: DZS, 1997.

2. Adleman, LM. On constructing a molecular computer [Web Page]. 1995; Available at ftp.usc.edu/pub/csinfo/papers/adleman/molecular_computer.ps.

3. Adleman, LM, Rothemund, PWK, Roweis, S, Winfree, E. On applying molecular computation to the Data Encryption Standard [Web Page]. Available at ftp://hope.caltech.edu/pub/pwkr/DIMACS/des.ps.

4. Alberts B, Bray D, Lewis J. Alberts B, Bray D, Lewis J. Molecular biology of the cell. 3 edition. New York & London: Garland Publishing, inc., 1994: 106,234.

5. Barnes WM. PCR amplification of up to 35-kb DNA with high fidelity and high yield from bacteriophage templates. Proc. Natl. Acad. Sci. 1994; (91):2216-20.

6. Blanchard AP, Kaiser J, Hood LE. High- density oligonucleotide arrays. Biosens. Bioelec. 1996; Vol. 11:687-90.

7. Brodnik A, Dobrin A, Drobnič M. Leksikoni Cankarjeve založbe - Računalništvo. Ljubljana: Cankarjeva založba, 1988.

8. Chee M, Yang R, Hubbell E. Accessinggenetic information with high-density DNA arrays. Science 1996; Vol. 274:610-4.

9. Dassen, JHM. DNA computing: Promises, problems, perspective [Web Page]. 1997; Available at http://www.wi.LeidenUniv.nl/~jdassen/Potentials.ps.gz.

10. Drašler J, Grabnar M, Kreft I. Genska kontinuiteta. 9 edition. Ljubljana: DZS, 1994: 31-60.

11. Fodor SPA, Read JL, Pirrung C, Stryer L, Lu AT, Solas D. Light-directed spatially addressable parallel chemical synthesis. Science 1991; Vol. 251:767-73.

12. Gehani, A, LaBean, TH, Reif, JH. DNA-based Cryptography [Web Page]. 1999; Available at http://www.cs.duke.edu/~reif/paper/DNAcrypt/crypt.ps.

13. Guarnieri F, Fliss M, Bancroft C. Making DNA Add. Science 1996; (273):220-3.

14. Gupta V, Parthasarathy S, Zaki MJ. Arithmetic and Logic Operations with DNA. 3rd DIMACS Meeting on DNA Based Computers.

15. Kiernan, V. DNA-Based Computers Could Race Past Supercomputers [Web Page]. 1997; Available at http://chronicle.com/data/articles.dir/art-44.dir/issue.../14a02301.ht.

16. LaBean TH, Butt TR, inventors. Methods and materials for producing gene libraries. U.S. # 5,656,467. 20 August 1997.

17. LaBean TH, Kauffman SA. Design of synthetic gene libraries encoding random sequence proteins with desired ensemble characteristics. Protein Science 1993; 2:1249-54.

18. LaBean TH, Winfree E, Reif JH. Experimental Prograss in Computation by Self-Assembly. 5th Annual DIMACS Meeting on DNA Based Computers.

19. LaBean TH, Yan H, Reif JH, Seeman N. Construction and Analysis of a DNA Triple Crossover Molecule. 1998.

20. Pease AC, Solas D, Sullivan EJ, Cronin MT, Holmes CP, Fodor SP. Light-generated oligonucleotide arrays for rapid DNA sequence analysis. Proc. Natl Acad. Sci. USA 1994; Vol.91:5022-6.

21. Reif JH. Local Parallel Biomolecular Computation. 3rd DIMACS Meeting on DNA Based Computers.

22. Roberts SS. Turbocharged PCR. Jour. of N.I.H. Research 1994; (6):46-82.

23. Roweis S, Winfree E. A Sticker Based Model for DNA Computation.

24. Rubin H, Klein J, Leete T. A biomolecular implementation of logically reversible computation with minimal energy dissipation. 4th Int. Meeting on DNA-Based Computing.

25. Staroba, J, Helfen, M, Nichols, O, Martin, B. DNA computing [Web Page]. Available at http://www3. hmc.edu/~mhelfen/bio/bio2.html.

26. Stinson DR. Cryptography, Theory and Practice. Boca Ratan: CRC Press, 1995.