

Nataniel Jhutti

DATABEHANDLING MED BIOMOLEKYLER

Kandidatavhandling i datavetenskap
Handledare: Prof. Ion Petre och Dr. Eugen
Czeizler
Institutionen för datavetenskap
Åbo Akademi

Åbo 2015

INNEHÅLLSFÖRTECKNING

1	INLEDNING.....	1
1.1	Fokus och syfte	1
1.2	Allmänt om DNA och proteiner	2
1.3	ssDNA & dsDNA struktur (Watson crick bonding)	4
1.4	DNA Hairpins	5
2	DATABERÄKNINGAR MED DNA.....	7
2.1	Betydelse av CRN	7
2.1.1	Sträng förskjutnings reaktions nätverk.....	7
2.2	CRN programmering	8
2.3	DNA Reaktionen	8
2.3.1	DNA Hybridiserings-reaktioner.....	9
2.3.2	DNA Enzym reaktioner.....	9
2.3.3	DNAzyme reaktioner	9
3	FORSKNINGSPROBLEM OCH LÖSNINGAR	10
3.1	Konstruering av AND – OR grindar	10
3.2	SDRN Detektorer	12
3.2.1	Läckor begränsar detektorernas känslighet.....	13
4	PRAKTISKA EXEMPEL OCH TILLÄMPNINGAR	15
4.1	Microsoft visual DSD	15
4.2	Exempel (entropi driven katalytisk grind)	16
4.3	DNA Nanoteknologi - Walkers	19
4.3.1	Icke-autonomt opererande walkers	20
5	DISKUSSION OCH SAMMANFATTNING	22
	REFERENSER.....	23

1 INLEDNING

Databehandling med hjälp av biomolekyler är ett tillvägagångssätt inom datavetenskapen och biologin för att processera data och tillföra information. Det handlar om att lagra, behandla och lösa kombinatoriska problem av data på molekylär nivå.

Det finns många fördelar med att använda detta tillvägagångssätt istället för konventionella silikondatorer. Den största fördelen är för tillfället att man kan utföra beräkningar i miljöer som man inte förut kunde, exempel på dessa är på havsbotten och i levande biologiska celler. En annan teknisk fördel är att biomolekyler tillåter oss att organisera materia på nano-nivå, och de kan naturligt kommunicera med biologiska ekosystem.

Orsaken till varför vi inte helt och hållet har bytt från silikondatorer till denna metod är att det ännu är relativt kostnadssamt angående algoritmisk komplexitet och lagringskapacitet, och komplicerade beräkningar på konventionellt sätt är i dagens läge mer effektivt.

På grund av den biologiska materians naturliga uppbyggnad, och vår omständiga förståelse av denna, har man nu kunna hypotisera möjliga framtida tillämpningar. Dessa tillämpningar kommer jag att bland annat behandla i denna avhandling. Målet med avhandlingen är att i teorin ge en översikt av hurdana biologiska materia som används, samt grunderna i hur man programmerar denna materia med hjälp av logiska grindar och DNA sträng förskjutnings system (eng. DNA strand displacement systems) . Jag kommer i praktiken att använda mig av ett verktyg som heter ”Microsoft Visual DSD” för att simulera molekylärberäkningar. Denna avhandling är riktad åt en person med en datavetenskaplig bakgrund.

1.1 Fokus och syfte

DNA, och nukleiska syror är överlag väldigt unika på många sätt. Man förstår nyckelegenskaperna hos nukleiska syror, såsom DNA väldigt bra. DNA kan lagra information i sig själv väldigt länge jämfört med konventionella hårddiskivor. DNA's tertiära (tredimensionella) struktur är mer förutsägbar än andra molekylers, såsom till

exempel proteiner. Detta är en väldigt viktig fördel eftersom utan denna förutsägbarhet skulle den inte tillhandahålla en struktur som fungerar. Deras hybridiserade reaktioner förstås också ganska bra, vilket gör att man kan förutspå hur de kommer att reagera med annorlunda, kombinerad DNA. Reaktionerna är också produktivt kontrollerbara bland annat med hjälp av ”Toeholds”.

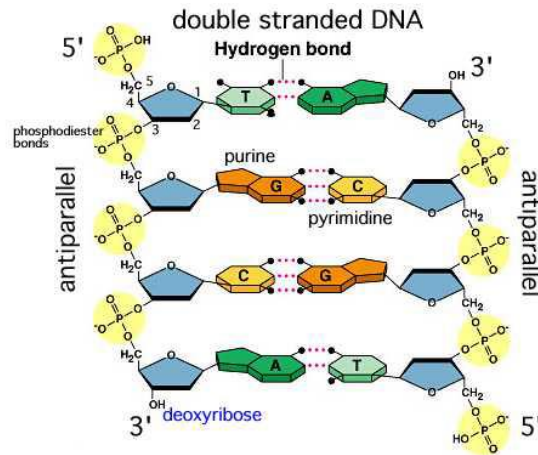
Det går att utföra en stor mängd operationer på DNA samtidigt. Konventionella datorer och elektroniska enheter har blivit förminskade med ”top-down” metoden (refereras också ibland till som nedbrytningsmetoden), och används för att bygga komplexa system [1], och metoden närmar sig sin gräns för att det inte längre ska vara kostnadseffektivt. Med detta menas att vi närmar oss en gräns där det inte längre lönar sig att göra enheterna mindre utan att prestandan lider. Molekylär databehandling, inkluderande nanoteknologi (se kap. 4.3) använder sig av en ”bottom-up” metod. Detta innebär att den själv-assemblerar inifrån ut, och är självuppehållande. Metoden använder sig av biologiskt bränsle, vilket tillger möjligheter som inte silikon-datorer kan ge idag, till exempel databehandling och lagring i organiska material, som i en cell, eller på havsbotten. Databehandling med biomolekyler möjliggör mätning av olika kemiska substanser, samt upptäckning och igenkänning av kemiska och biologiska händelser på cellnivå [2].

1.2 Allmänt om DNA och proteiner

Före DNA databehandling och datalagring diskuteras så måste det tillhandahållas en viss bakgrundsinformation om DNA, proteiner, nukleotider samt föreningen av dessa. DNA är en molekyl som kodar genetiska instruktioner, de är i sin tur utnyttjade i utvecklingen och beteendet av alla levande organismer (utom i RNA virus) [2]. DNA är ibland refererat till som en manual eller en instruktionsbok för organism utveckling. Proteiner, kolhydrater och DNA formar alla entydiga mängder nukleiska syror. Dessa tre typer av syror formar grunden till allt liv [2].

DNA är en polymer som existerar i endera singelsträng eller dubbelsträng (dubbelhelix). Dessa strängar består av upprepande monomer, som i sin tur kallas nukleotider. Nukleotiderna innehåller tre stycken olika kemiska komponenter, en 5' kol-socker molekyl, en kvävebas och en fosfatgrupp. Fosfatgruppen är fäst vid en kol atom C5 på ena änden, och vid C3 på motsatta änden. Denna struktur gör att DNA strängen får en

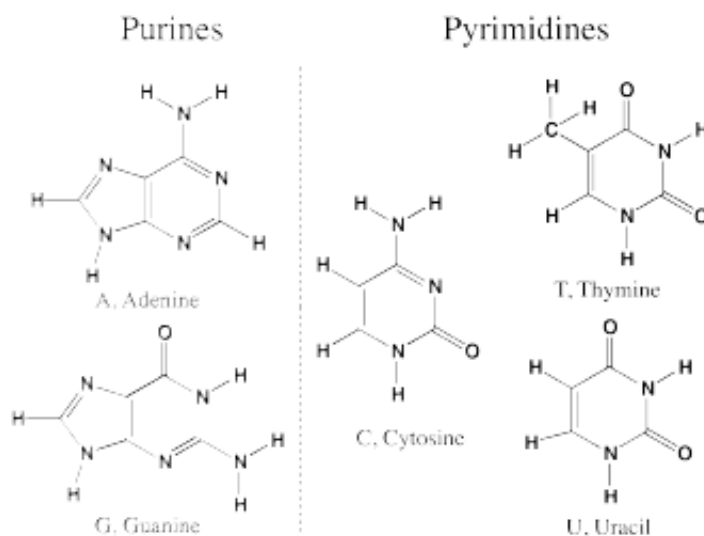
tydlig riktning. Dessa ändor refereras till som "5'" (den primära änden) och "3'" (motsatta änden). Komponenterna formar kärnfunktionen (eng. "backbone") för DNA strängen (se Figur 1) [2].



Figur 1: Dubbelsträngad DNA [2].

Det finns också flera komponenter (baser) i en nukleotid som inte hör till kärnuppbyggnaden av en DNA sträng. Det existerar fem stycken av dessa baser, Adenine(A), Guanine(G), Cytosine(C), Thymine(T) och Uracil (U). Uracil finns dock ej i DNA nukleotider [3].

Baserna A och G hör till en grupp (klass) som kallas Puriner, medan C, T och U tillhör en grupp som kallas Pyrimidiner. Detta är väldigt viktigt att veta i kapitel (1.3: ssDNA & dsDNA struktur) när dubbelsträngad DNA behandlas. Nedan i Figur 1.1 visas skillnaden mellan strukturen hos Puriner och Pyrimidiner. Största skillnaden mellan dessa är att en Purine innehåller 4 stycken kväveatomer och pyrimidine 2 kväveatomer.



Figur 1.1 Puriner och Pyrimidiner [2]

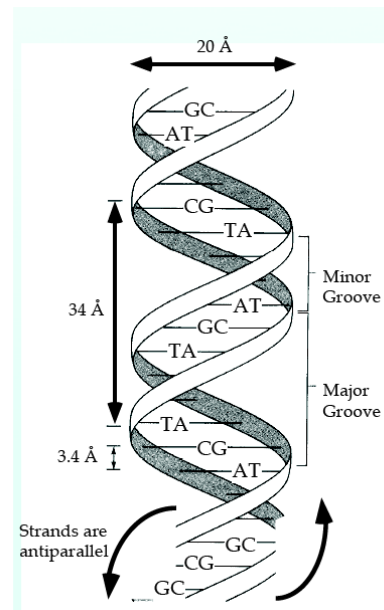
1.3 ssDNA & dsDNA struktur (Watson crick bonding)

DNA existerar i två olika former, singel sträng (ssDNA) och dubbelsträng (dsDNA). När DNA är i sin singel strängad form, så kan den ses som en lång sekvens av bokstäver. I många fall formar den en sekundär struktur där strängen slingar runt sig själv och formar vätebindningar med andra komponenter i sig själv, detta fenomen kallas för ”*random coil*” [2].

dsDNA formen bildas så att två stycken ssDNA binder sig till varandra (hybridiserar, se kapitel 2.3.1) och skapar en dubbelhelix. Innan ssDNA strängarna binder sig till varandra så svänger de sig antiparallelt (se figur 1.2), det betyder att ena strängen har en 5' till 3' riktning och andra strängen en motsatt riktning, 3' till 5' (se kapitel 1.3).

Vissa komponenter i DNA strängarna binder sig alltid till varandra vid skapelse av dsDNA, denna binding kallas för ”Watson Crick bonding”. Adenine binder sig alltid till Thymine, och Guanine binder sig alltid till Cytosine. Märk väl att Puriner alltid binder sig till Pyrimidiner [2].

När dessa är bundna, kommer det att finnas två stycken linjära sekvenser av bokstäverna A,T,G och C. Sekvenserna är komplementära med varandra och passar in i varandra som nyckeln i ett lås, så ifall man vet sekvensen av den ena strängen så kan man logiskt lista ut den andra. Dessa sekvenser kan ses som DNA instruktioner, och kan med lite modifiering omvandlas till binära tal [3].

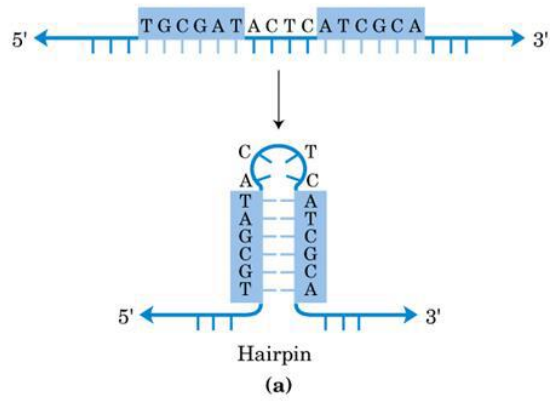


Figur 1.2 dsDNA dubbelhelix [2]

1.4 DNA Hairpins

En DNA Hairpin innebär en sekundär struktur formad av en ssDNA. Strukturen innehåller en dubbelsträngad region, en stam och en ohybridiserad slinga [4]. Hairpinen kan endera vara öppen som en linjär tråd, eller stängd (se Figur 1.3). En teknik för att öppna slingan i en hairpin kallas ”toehold mediated strand displacement” som kommer att diskuteras i kapitel 2.3. Orsaken till att Hairpins behandlas i avhandlingen är att de är väldigt användbara strukturer inom datavetenskapen, eftersom de har stor databehandlings-kapacitet. De största fördelarna är: (1) att de lagrar en stor mängd energi, vilket avges under sling-hybridisering (processen där man kombinerar två stycken strängar till en enda sträng). När energin avges så orsakar den en reaktion, som driver behandlingsprocessen framåt. Energin fungerar som bränsle för databehandlings-processen.(2) När de är i Hairpin form, så reagerar de inte mycket till andra (icke slingade) DNA strängar. Detta gör att de kan fungera som monomer. Deras beteende kan också effektivt kontrolleras på grund av detta [2].

Till implementeringsområden av DNA Hairpins hör: skapelse av AND-OR logiska kretsar, energitillhandahållare för formning av stora polymerer och för programmeringsstigar inom DNA själv-assemblering.



Figur 1.3 DNA Hairpin [4]

2 DATABERÄKNINGAR MED DNA

2.1 Betydelse av CRN

CRN (Kemiska reaktions nätverk) är verktyg för förståelse av DNA beräkningar, samt ett hjälpmedel för att förstå molekylär programmering. Betydelsen av denna är alltså uppdelad i två roller, att stå som en modell för analysering och kvantifiering av specifika DNA beräkningar, och ett programmeringsspråk för att beskriva informationsflödet i molekyler [2]. Den första av dessa roller är inte unik för datavetenskapen på något sätt, men är ett väldigt centralt begrepp inom biologin för att beskriva biokemiska processer.

Detta har nu kombinerats med datavetenskapen och sålunda har rollen för programmerings-språket skapats. Med hjälp av detta programmeringsspråk har DNA kunnat börja fungera som ett medie för informationsflöde. Tack vare detta kan man dator-emulera de flesta nedritade CRN modeller. Det tillåter oss också att modifiera informationsflödet, och effektivt kunna testa denna.

2.1.1 Sträng förskjutnings reaktions nätverk

Sträng förskjutning (eng ”Strand displacement”) är en process, där två DNA strängar med full eller partiell komplementaritet hybridiserar till varandra. Den inkommande strängen binder sig först till en toehold(se kapitel 2.3.1), och sedan byter ut (kontrapositionerar) den utgående strängen med ett tillvägagångssätt som kallas ”One-dimensional random walk process” [2]. När många DNA strukturer utför detta på samma gång så uppstår det ett nätverk av reaktioner. Dessa reaktioner refereras till som sträng förskjutnings reaktions nätverk, eller ”SDRN’s”.

SDRN’s har unik utformning, denna unikheter tillåter dem att bli modellerade som CRN’s. Detta tillför många möjligheter, som att kunna lära sig förutspå en potentiell användbar CRN med hjälp av utföring av experiment. Det finns dock vissa restriktioner angående antalet reaktioner som kan uppstå. Vissa av dessa reaktioner har visat sig vara enkla att förutspå på basis av tidigare erfarenhet och olika variabler såsom Termodynamiska parametrar(eng. Thermodynamic parameters). Parametrarna mäter

bland annat temperaturen på olika nukleotider, och på basis av det definierar stabiliteten hos dem [5].

Enzym fria DNA – beräkningsenheter kan exekveras inom flera olika koncept, till dessa hör booleska kretsar och neurala nätverk, förstärkare av nukleiska syror, ändliga automater (eng. finite state machines) och molekylära walkers. Dessa är alla exempel på sträng förskjutnings reaktions nätverk, och en del av dessa kommer att diskuteras senare i denna avhandling.

2.2 CRN programmering

I kapitel 2.1 (Betydelse av CRN) nämnde jag att CRN har två olika tillämpningssätt, det första är centralt inom biologin, och det andra för datavetenskapen. Programmeringstillämpningen låter oss illustrera hur (nästan) vilken CRN som helst kan bli översatt till en SDRN. Orsaken till varför vi är intresserade av detta är att på låg-nivå emulera hur en CRN beter sig. Man kan tänka sig CRN som ett konventionellt högnivå programmeringsspråk, och SDRN som ett mer lågnivå språk [2].

Orsaken till varför vi är intresserade av att konvertera CRN till SDRN är jämförbart med fördelarna med att använda ett konventionellt högnivåspråk. CRN är mer abstrakt, och därför enklare att konstruera. CRN är ett väldigt kraftfullt programmeringsspråk, det kan uppnå turing-universell beräknings-standard med ganska lite felberäkningar. Dock helt felfria databeräkningar är inte möjliga. Man kan endast beräkna semi-linjära funktioner utan felaktighet [2].

2.3 DNA Reaktionen

För att vi ska kunna förstå hur data kan processeras och beräknas inom en DNA molekyl, så måste vi vara medvetna om egenskaperna hos DNA, och hur de reagerar med varandra när de binds och lösgörs med varandra. Dessa reaktioner kan grupperas i tre stycken underkategorier: (1) DNA hybridiserings reaktioner: till denna underkategori hör ”toehold mediated strand displacement” som nämnt i kapitel 1.4 , (2) DNA enzym reaktioner, (3) DNAzyme reaktioner [2]. Dessa olika kategorier kommer att diskuteras i mer detalj i detta kapitel.

2.3.1 DNA Hybridiserings-reaktioner

Watson Crick bonding, är ett väldigt bra exempel på DNA hybridisering. Det innebär att två ssDNA binds till varandra och formar dsDNA. Alla tillfällen då DNA strängar binds till -eller lösgör sig från varandra, refereras till som en DNA hybridiseringsreaktion. När en dsDNA utsätts för en tillräckligt hög temperatur, kan de två strängarna lösgöra sig från varandra. Denna temperaturgräns kallas för *smälttemperaturen*(eng. *Melting temperature*), och definieras strikt som temperaturen där 50 % av dsDNA konverteras till ssDNA [2].

Termen ”Toehold mediated strand displacement” hör också till denna kategori. En ssDNA sträng kan fungera som ett signal-element när den inte är hybridiserad till någon annan DNA sträng. Termen signal-element betyder att den kan ta emot en signal för att behandla information på något sätt. En dsDNA kan inte ta emot eller behandla signaler, för att lösa detta finns det ett domän i dsDNA, kallad för ”Toehold”. Toeholden är ssDNA baserad och kan därför ta emot signaler från andra DNA strängar. Med detta förfarande kan man på basis av Toehold längden kontrollera reaktionshastigheten för en enskild databehandling [6].

2.3.2 DNA Enzym reaktioner

Enzymer är en form av proteiner som ökar eller minskar hastigheten på olika biokemiska reaktioner. Enzymer har potentiellt väldigt hög effektivitet och snabbhet. De flesta medier inom en biologisk cell behöver enzymer för att försnabba nedbrytningsprocesser [2].

2.3.3 DNAzyme reaktioner

DNAzyme står för ”deoxyribozym” [7], forskare har under senaste tid använt sig av denna samt aptamerer för att katalysera DNA hybridiseringsreaktioner. En aptamer är en nukleotid eller peptid som binder sig till en specifik molekyl. En DNAzyme är isolerad *in vitro* medan aptamerer är upptäckta och isolerade *in vivo*. Fördelen med denna metod är att deras enzymatiska aktivitet överstiger den av Watson Crick hybridiseringens. De har också väldigt specifika bindningar till vissa molekyler, vilket kan utnyttjas i viss sorts implementering. En av dessa implementeringar är inom cancerforskning, eftersom cancerceller alltid är av en specifik typ [7];[2].

3 FORSKNINGSPROBLEM OCH LÖSNINGAR

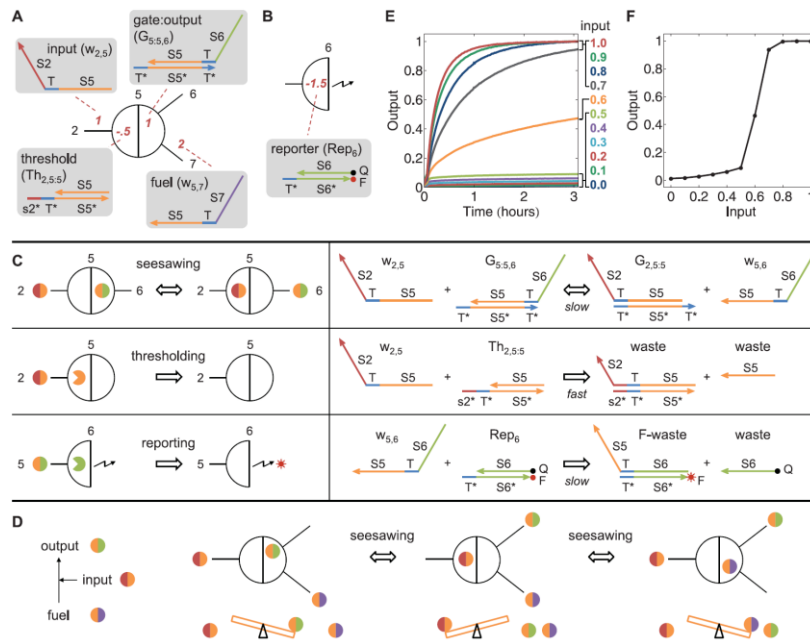
Fastän forskningen har kommit långt angående databehandling med biomolekyler, så finns det ändå vissa problem angående implementationen i praktiken. Till dessa problem hör exempelvis konstruktion av DNA kretsar. Faktumet att det finns flera miljarder DNA strängar till hands som förväntas att samarbeta, gör detta till en väldigt svår sak att utföra. De förväntas först att bli inmatade med data, varav de sedan måste behandla datan på ett entydigt sätt så att förväntat resultat nås. För att uppnå denna standard, har man skapat olika sorters logiska grindar och kretsar med användande av olika DNA-motiv [2]. Dessa olika sorters kretsar kommer vi att behandla i detta kapitel. Vi kommer även att diskutera för och nackdelarna med diverse sorters kretsar. Det är också värt att nämna att det finns två olika huvud-tillvägagångssätt angående implementation och konstruering av dessa kretsar, nämligen logiska grindar och kemiska reaktionsnätverk. Dessa kommer vi också att diskutera i mer detalj [2].

3.1 Konstruering av AND – OR grindar

Det finns två olika sorters sträng förskjutnings - designade DNA kretsar, lösningsbaserade och lokaliserade kretsar. Den lösningsbaserade kretsen grundar sig på följande: DNA kretsar innehåller en väldigt stor mängd sammanhängande molekyler, och varje molekyl reagerar med en specifik delmängd av molekyler. De kretsar där molekylerna sprider ut sig genom en lösning för att hitta den specifika delmängden molekyler så kallas för *lösningsbaserad krets* [2]. AND och OR logiska grindar grundar sig på en sådan krets. Grindarna grundar sig på David yu Zhangs och Erik Winfrees [6] Toehold protokoll, som förklaras noggrannare i kapitel 2.3.1.

Lulu Quan och Erik Winfree [6] har utvecklat en skalbar, **lösningsbaserad** krets-metod där idén är en seesaw grind grundad på Toehold protokollet. Metoden består av en mängd seesaw grindar, och varje grind kan acceptera en inmatning (DNA sträng) och producera ett resultat (DNA Sträng). Varje sträng som matas in fungerar katalytiskt, vilket innebär att en inmatning kan producera flera olika resultat från flera olika seesaws samtidigt. Detta är ibland refererat till som parallella databeräkningar (eng. Parallel computing).

Istället för hög och lågspänning som används i konventionella datorer, använder de sig av biologiskt bränsle. Man kan tänka sig inmatningen som bränslet. Resultat-strängarna fungerar som inmatnings-strängar i nästa krets-beräkning, beräkningarna utförs i omvänd-kronologisk ordning. Komponenterna i ett seesaw system, samt hur det fungerar demonstreras i figur 3.1.



Downloaded from www.sciencemag.org on June 4, 2011

Figur 3.1, Komponenter i ett seesaw system[2].

Strängen för inmatning har 3 stycken domän, S2, T och S5 (figur 3.1, del A). Strängen för resultat (utmatning) är S5,T,S6. Domän S2 och S5 jämförs sedan vid utmatnings/resultats grinden (output:gate). Detta förfarande utnyttjas i de logiska AND – OR kretsarna.

En OR grind baserar sig på en väldigt simpel princip, det finns åtminstone två uttryck (utmatningar) där minst ett av resultaten stämmer överens med inmatningarna. Ifall inte, evalueras uttrycket till falskt. I biomolekyllära beräkningar är principen samma, minst ett domän av utmatningen(x,y) måste stämma överens med inmatningen (x,y). Ifall inte, evalueras resultatet till falskt. Ta följande som exempel: $x = S5,T,S6$ och $y = S6,T,S6$. Ifall nåndera av dessa ger ett resultat i utmatnings-strängen så evualueras grinden till TRUE. En AND grind är också relativt simpel i teorin, den kräver att båda uttrycken (x,y) stämmer överens med utmatningen, och isåfall evualuerar uttrycket till TRUE.

I figur 3.1 visas ”threshold gate”. Den har blivit introducerad av författarna Lulu Qian och Erik Winfree [6] för att lagra en DNA sträng efter att den blivit evaluerad av en AND-OR logisk grind. Man kan också tänka på denna grind som en sopsamlare, eftersom den isolerar en godtycklig sträng och gör den oanvändbar. Orsaken till att denna ”threshold gate” har skapats, är att kunna använda sig av AND-OR grindarna för att skapa vilket logiskt uttryck som helst, utan att samla upp onödiga resultatsträngar [2].

På grund av att vi nu kan skapa vilket logiskt uttryck som helst, kan vi skapa en godtycklig krets och beräkna ett önskat resultat med hjälp av dessa grindar. Ett exempel kunde vara att skapa en krets som beräknar kvadratroten av $n \leq 15$. Detta lyckas med att koppla grindarna följdriktigt. Kretsen som är utformad av Qian och Winfree, tar in 8 stycken inmatningar. Inmatningarna representerar binära tal, alltså en 0 eller 1 för varje bit. Hur bitarna definieras för de binära talen beror på resultatet som beräknas. Utmatnings-strängen utmärker sedan den mängden bitar som är representerade som 1:or, och på det sättet beräknar resultatet.

Det finns dock vissa nackdelar med denna metod. $O(N)$ Komplexiteten ökar linjärt med djupet på kretsen, och det är därför en väldigt långsam beräkningsmetod. Det kan ta upp till 10 timmar att beräkna kvadratroten av exemplet som nämnts tidigare. För att kunna göra snabbare beräkningar med pålitliga resultat, så använder man istället andra slags kretsar som kommer att diskuteras senare i detta kapitel.

Detta är ett exempel på en beräkning som utförs in vitro. Det betyder att det utförs i ett medium som ligger utanför en levande organism, som till exempel ett provrör. Den identifierar också alla inmatnings-strängar som finns i närheten, detta är en stor fördel [6];[2].

3.2 SDRN Detektorer

En stor utmaning inom detta forskningsområde är att konstruera en lösningbaserad krets för att känna igen en specifik DNA art (eng. species) som blivit utmatad av en molekyl. För att detta ska lyckas så måste inmatningen monitoreras för korrekt mängd inmatning. Ifall inmatnings-kvantiteten är för liten, kommer inte utmatningen att kunna mätas korrekt, vilket leder till fel resultat. Därför måste detektorn vara såpass känslig att den

också känner igen små mängder inmatning. Inmatnings-mängden är alltid specifik beroende på vilken typ av CRN som behandlas. Detektorn måste därför kunna anpassa sig till olika temperaturer, koncentreringar och olika buffert villkor.

En SDRN detektor, också refererad till som ”polymerase chain reaction” eller ”PCR” är en enzym baserad detektor. Det är ett mät-instrument, som kräver temperaturväxling och drivs av en väldigt bred och komplex mängd enzymer. Tack vare framgången inom detta vetenskapsämne kan en PCR i dagens läge känna igen så mycket som 2000 DNA molekyler i bara 1 mL lösning [2].

Det existerar också andra slags enzym baserade detektorer, men de förlitar sig till en betydligt mindre mängd enzymer och innehar inte alla egenskaper som PCR har. Därför är PCR exceptionellt unik och väldigt användbar. Den är också väldigt enzymatiskt modifierbar, vilket gör att den kan känna igen olika sorters sekvenser i olika omgivningar relativt enkelt. Eftersom den har dessa egenskaper, så kan den också operera som en subrutin i en annan (komplex) SDRN.

Den största nackdelen med PCR är som sagt att den inte känner igen tillräckligt små mängder inmatning. För detta att lyckas så måste PCR ha ett specifikt särdrag som tillåter den att förstärka en tillräckligt liten inmatnings signal till en tillräckligt stark utmatningssignal. Förstärkningseffekten måste också kunna stängas av helt och hållet i avsaknad av inmatning, vilket inte sker på grund av *läckor* (se kapitel 3.3.1).

Detta problem hoppas forskare att inte är oöverkomligt. Hittills har man endast utfört databeräkningar med detta tillvägagångssätt in vitro. Igenkänning av mindre mängd inmatning skulle tillåta oss utföra beräkningar in vivo [2].

3.2.1 Läckor begränsar detektorernas känslighet

Termodynamiken säger oss, att ifall en kemisk reaktion inträffar i närvaro av en katalysator(aktiverare), så måste reaktionen också inträffa i frånvaron av en, oftast dock i mycket mindre hastighet och skala. Detta är oönskat beteende, och leder till såkallade *läckor*(eng. *leaks*) [2]. Dessa läckor orsakar ett oönskat resultat i sådana situationer där

ingen inmatning har getts. Identifiering och förmildrande av dessa läckor är nyckeln till att förstå hur man ska implementera en känslig detektor.

För enkla CRNs, kan man identifiera läckor väldigt enkelt. Detta är på grund av den breda förståelsen av katalytiska reaktioner hos dem. För mer avancerade CRNs är det dock mycket svårare att identifiera katalytiska reaktioner.

Eftersom det går att identifiera läckor hos enkla CRNs på basis av en grundlig förståelse av alla dess reaktioner, säger det oss att nyckeln till att begränsa läckor hos SDRN detektorer är samma som hos enkla CRN's [2].

4 PRAKTISKA EXEMPEL OCH TILLÄMPNINGAR

I detta kapitel kommer jag att behandla olika slags praktiska tillämpningar inom databehandling med biomolekyler. Jag kommer att introducera en mjukvara (Visual DSD) för att implementera CRN programmerings-språket. Jag kommer också att illustrera ett implementations exempel med användning av Visual DSD.

Till följd av utvecklingen av modifierbara lösningsbaserade kretsar (CRN), och DNA sträng förskjutnings tekniker har forskare kunnat utveckla en ny teknologi kallad nanoteknologi. DNA nanoteknologins grunder ligger i modifierbara DNA strängar. Databeräkningar har hittills utförts mestadels in vitro, men också in vivo, mer specifikt i kackerlackor(*Blaberus discoidalis*) [9]. Användningen av denna teknologi är speciellt användbar inom den medicinska industrin. Forskare har förutspått att man i framtiden kommer att kunna utföra lokala läkemedels doseringar, samt utföra mindre kirurgiska ingrepp in vivo med hjälp av nano-roboter. Denna teknologi kommer jag också att berätta mera om i detta kapitel.

4.1 Microsoft visual DSD

Visual DSD är en programvara utvecklad av Microsoft för att i teorin simulera kemiska reaktions nätverk samt logiska kretsar. Det är konstruerat för att hjälpa forskare i utvecklingen av olika slags molekylära enheter, och utför sitt ändamål väldigt bra, eftersom den tillger fördelarna av att kunna visualisera en implementerad CRN utan att manuellt behöva bygga reaktionsnätverket [10]. Det är i grund och botten en implementation för ett programmeringsspråk för modifierbara DNA kretsar. Användaren av programvaran behöver inte vara expert i området för att kunna använda det. Programmeringsspråket inkluderar alla grundläggande element för domänsekvenser och toeholds. Programvaran stöder inte någon möjlig sekundär struktur av en DNA sträng (se kapitel 1.4 om DNA Hairpins).

Visual DSD verktyget kompilerar en mängd DNA molekyler till en mängd kemiska reaktioner. Den tillger snabb prototypning av CRN och beräknar automatiskt alla reaktioner mellan DNA arter. Dessa reaktioner kan sedan simuleras visuellt i form av olika slags grafer och tabeller [10].

I kapitel 4.2 kommer jag att visa ett exempel på implementering av en simpel CRN med användande av Visual DSD.

4.2 Exempel (entropi driven katalytisk grind)

Visual DSD mjukvaran har många inbyggda exempel av CRN simulationer. För att nämna två väldigt kända exempel: CRN implementation av Lotka-volterra modellen som är en simpel implementation av en predator-bytesdjursekvation, och en liknande implementation av en entropi driven katalytisk grind (eng. "Entropy-driven catalytic gate") [11]. I detta exempel kommer jag att illustrera en implementation av den entropi drivna katalytiska grinden.

I kapitel 3.1 om AND-OR kretsar, nämndes det lite om katalytiska grindar. De refererades till som "seesaw grindar" och har möjligheten till att förstärka en inmatningssignal. De möjliggör också parallella databeräkningar.

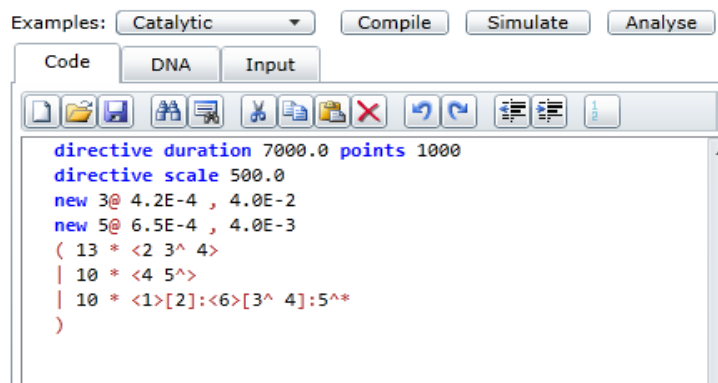
Varje katalytisk grind kan abstrakt representeras som en nod, med en eller flera trådar kopplade till vänster och höger sida. Varje tråd korresponderar till en ssDNA molekyl med en specifik sekvens. Denna tråd är också refererad till som en signal sträng (eng. "signal strand"). Varje nod korresponderar till en grinds ssDNA sträng, som utlöser signal strängar på begäran. När många av dessa katalytiska grindar aktiveras på samma gång, förstärks inmatningssignalen och på samma gång ökar beräknings komplexiteten [11].

Vi börjar med att se på programkoden, i detta exempel kommer inte syntaxen av programmerings-språket alls att gås genom, endast en kort genomgång och visualisering av hur detta har implementerats.

"Code" fältet visar själva programkoden, se figur 4.2.1; de två första raderna indikerar endast varaktigheten av själva simulationen, och den molekylära storleksskalan som systemet använder för att "skala upp" från molekylär nivå till simulering. De nästa två raderna indikerar två domän, som innehar specifika DNA sträng - bindnings och lösgörings hastigheter. Den sista kodraden i programmet representerar en mängd DNA molekyler, och deras specifika art koncentrationer. Fliken som är markerad med "Input" visar den grafiska representationen av "Code" fliken.

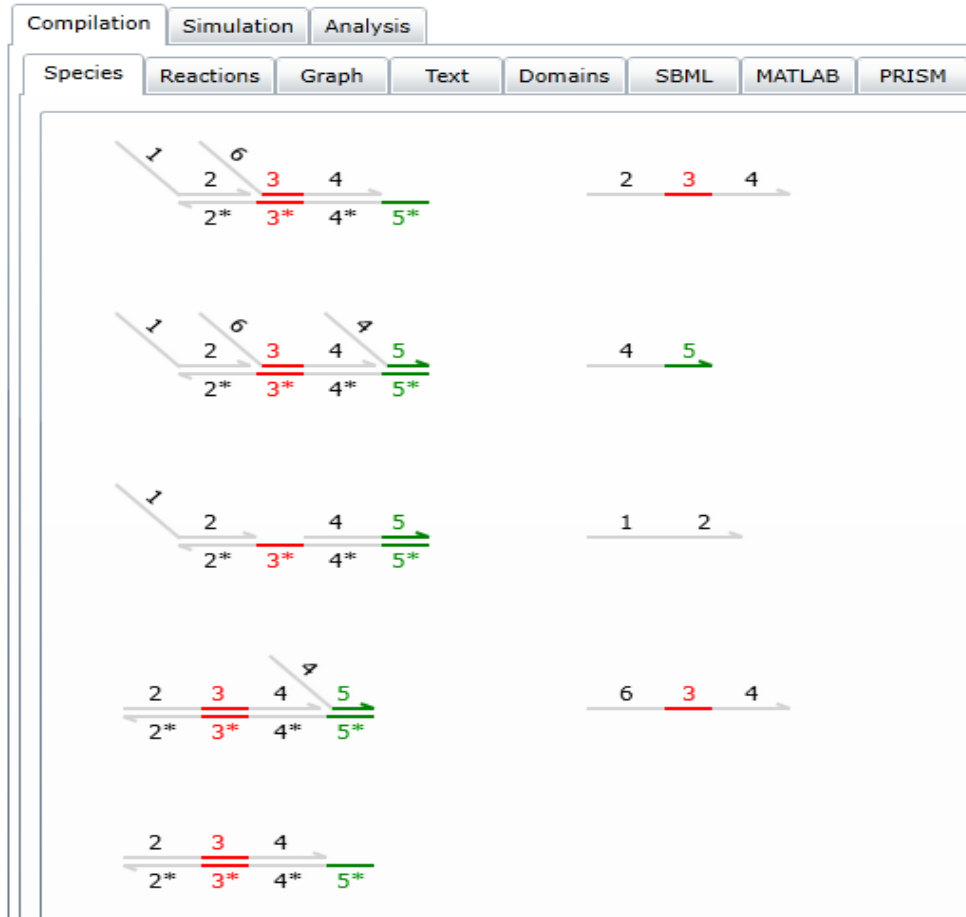
För att kompilera programmet, trycker man på ”Compile”. Den kommer att översätta programkoden till kemiska reaktioner. När programmet har kompilerats kommer det att uppstå nya flikar brevid ”Input” fliken. Dessa flikar visar, från diverse aspekter, hur de kemiska reaktionerna utförs. I figur 4.2.2 visas de olika DNA arterna. Strängarna till vänster representerar dsDNA (parallella strängar) och strängarna till höger ssDNA (enkel sträng). Figur 4.2.3 visar alla möjliga reaktioner mellan DNA arterna.

Figur 4.2.4 visar i ”DNA” fältet de olika toehold sekvenserna (se kapitel 2.3.1). I ”Text” fältet visas information om de olika arterna samt mängden av alla möjliga kemiska reaktioner som kan uppstå. Dessa är de mest relevanta egenskaperna programmet har för att simulera en CRN, dock så finns det diverse flera representationer och finjusteringsval men dessa kommer inte att gås igenom i denna avhandling.

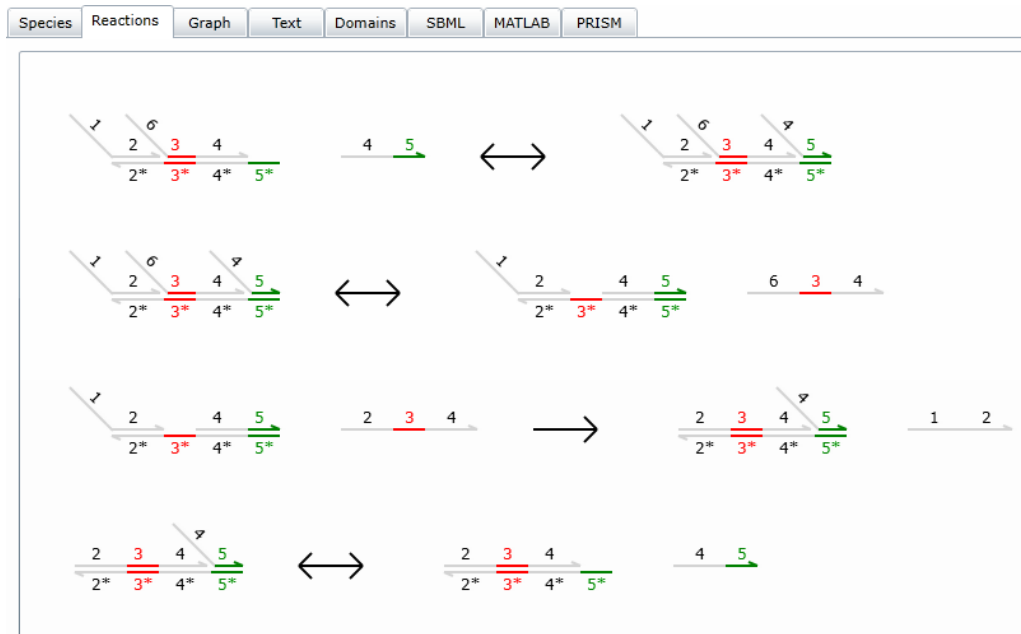


```
directive duration 7000.0 points 1000
directive scale 500.0
new 3@ 4.2E-4 , 4.0E-2
new 5@ 6.5E-4 , 4.0E-3
( 13 * <2 3^ 4>
| 10 * <4 5^>
| 10 * <1>[2]:<6>[3^ 4]:5^**
)
```

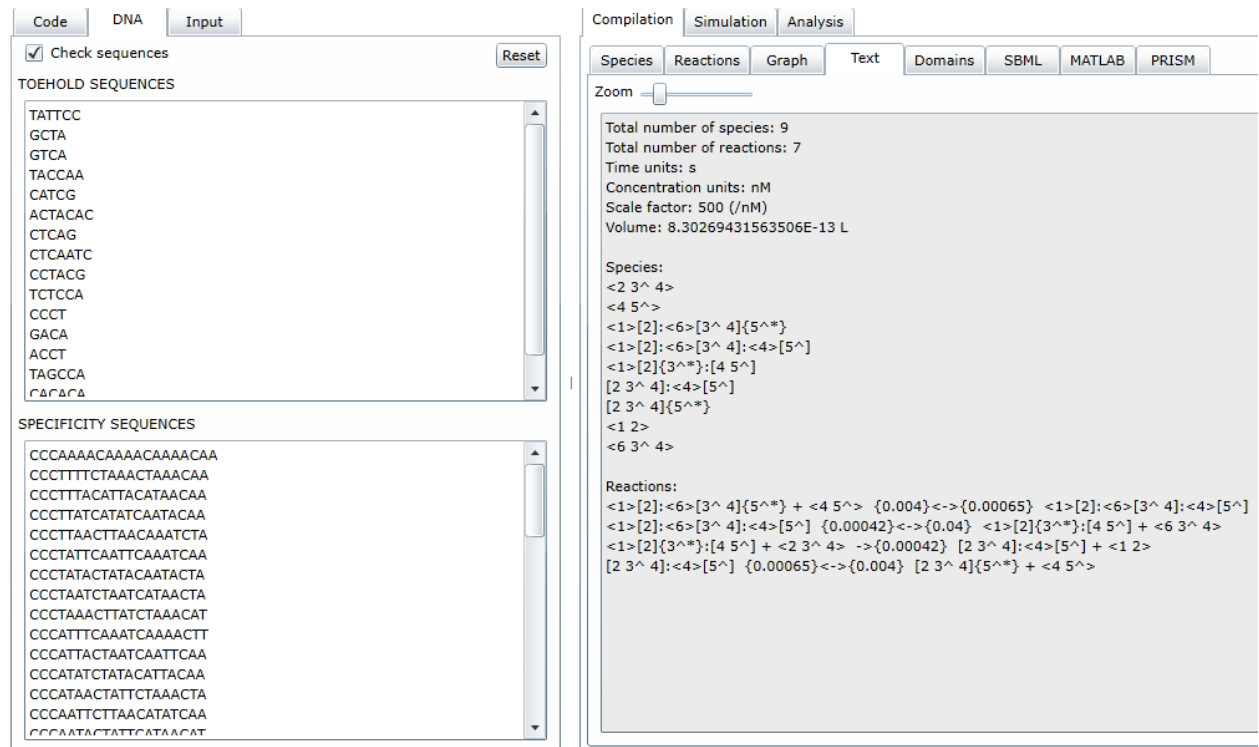
Figur 4.2.1



Figur 4.2.2



Figur 4.2.3



Figur 4.2.4

4.3 DNA Nanoteknologi - Walkers

Nanoteknologi är konstruktionen och uppbyggnaden av artificiella nukleiska syror för teknologiskt bruk. Specifikt i DNA nanoteknologi använder man DNA som behandlings-medie. Huvudmålet i DNA nanoteknologi är att syntetisera molekyllära motorer som är programmerbara och kapabla av att operera självständigt. Med molekyllära motorer menas olika slags självgående proteiner som innehar essentiella funktioner för biologiskt liv, såsom proteiner från Dynein, Kinesin och Myosin familjerna [9];[2].

Orsaken till varför DNA används för denna teknologi är att det är ett relativt enkelt att använda som byggnadsmaterial för dessa syntetiska motorer, eftersom DNA's uppbyggnadsstruktur och komponenter är väl igenkända.

En walker är ett syntetiserat DNA system som innehar förmåga att röra sig i substrat. En stor utmaning för forskare är att konstruera dessa walkers på ett sådant sätt så att de är kapabla av att operera självständigt på nano-nivå. Med detta menar jag att de ska kunna vara programmerbara till att utföra specifika funktioner på ett självständigt sätt, så att de

inte behöver hjälp från externa medier[2]. De bör vara konstuerade på ett sätt så att de slutar fungera när bränslet i dem är utarmat.

För att nå detta mål, har forskare konstant försökt ändra på metoden som walkern rör sig på för att fullt förstå dens beteende, för samma orsak har de också försökt ändra på bränslekällan som walkern använder. Till följd av dessa experiment har forskare kunnat visa deras potential att transportera last (exempelvis läkemedel), deras förmåga att kunna röra sig längs en specifik programmerbar stig, och förmågan att utföra kontrollerade kemiska reaktions nätverk.

Man kan dela in dessa walkers in tre kategorier på basis av experiment utförda av forskare. Dessa är 1) graden av förmåga att röra sig självständigt, det vill säga deras förmåga att röra sig autonomt 2) deras bränsletyp, eller 3) båda [2]. I nästa kapitel kommer jag att diskutera den första av dessa olika kategorier.

4.3.1 Icke-autonomt opererande walkers

I kapitel 4.3 nämnde jag att walkers kan delas in i tre kategorier, i detta kapitel behandlar jag den först-nämnda.

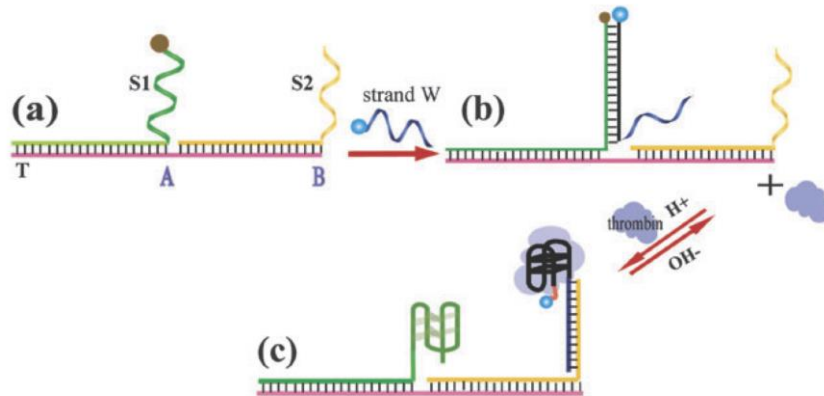
Nanomaskiner som fungerar på basis av förändringar i dess miljö kallas icke-autonoma nanomachiner, eftersom de är väldigt beroende av externa medier. Till dessa medier tillhör: förändringar i pH-värdet och salt concentrationen i lösningen, och introduktionen av faktorer som ändrar på den syntetiska DNA's funktion [2].

Jag vill illustrera detta med ett exempel som Wang et. Al demonstrerade [8];[2]. Han introducerade en walker som konstant ändrade läge på ett spår(se figur 4.1). Varje läge består av en ssDNA som är delvis hybridiserad till spåret. Denna ssDNA har en lös/hängande region som fungerar som en bindande plats för walkern. De olika lägen har olika pH-balanser, det första läget är uppbyggt av en i-motiv DNA sträng (eng. "i-motif DNA strand") [2] som kollapsar på sig själv och drar in hängande regionen under förekomst av låga pH-värden (pH 5). Under förekomst av högre pH-värden (pH 8) så ändrar den tillbaka till sitt ursprungliga tillstånd.

Resultatet var att när walkern var introducerad till spåret, hybridiserade den till första regionen ifall lösningens pH värde var högt, annars hybridiserade den till nästa region

som hade högt pH-värde, det visade sig att under förekomst av låga pH-värden kunde walkern inte binda sig till en region [2].

En stor nackdel med denna konstruktion är att walkern endast kunde röra sig mellan två stycken regioner, och kan därför inte ta vidare framsteg.



Figur 4.1, Molekylär walker [2]

5 DISKUSSION OCH SAMMANFATTNING

REFERENSER

- [1] <http://www.isi.edu/~lerman/papers/methodRevisedFinal.pdf>; Top-Down vs Bottom-up Methodologies in Multi-Agent System Design Valentino Crespi, Aram Galstyan, Kristina Lerman
- [2] Computing handbook set – Computer science vol 1, Sudhanshu Garg, Reem Mokhtar, Tianqi Song, Hieu Bui, Nikhil Gopalkrishnan, John Reif
- [3] http://www.biochemistry.org/Portals/0/Education/Docs/BASC02_full.pdf; Top-Down vs Bottom-up Methodologies in Multi-Agent System Design Valentino Crespi, Aram Galstyan, Kristina Lerman
- [4] https://books.google.fi/books?id=DVwsMvV7KcQC&pg=PA292&lpg=PA292&dq=dna+hairpins+computation&source=bl&ots=OxCvA3DLBF&sig=IyUdjaggff-DICSHsVhzhCYrE0&hl=en&sa=X&ei=O_8HVbmxDsTMyAP214GgBA&ved=0CFAQ6AEwBg#v=onepage&q=dna%20hairpins%20computation&f=false; Computing with Hairpins and secondary structures of DNA, Masami Hagiya, Satsuki Yaegashi, and Keiichiro Takahashi, p.295-307
- [5] <http://www.ncbi.nlm.nih.gov/pubmed/7545436>; Thermodynamic parameters to predict stability of RNA/DNA hybrid duplexes, Sugimoto N, Nakano S, Katoh M, Matsumura A, Nakamuta H.
- [6] <http://www.sciencemag.org/content/332/6034/1196.full#ref-9>; Scaling up Digital Circuit Computation with DNA Strand displacement Cascades, Lulu Qian, Erik Winfree
- [7] <http://mct.aacrjournals.org/content/7/2/243.full>; Molecular Cancer Therapy: cleave and let die, Crispin R. Dass, Peter F.M Choong and Levon M. Khachigian
- [8] Chunyan Wang, Jingsong Ren, and Xiaogang Qu. A stimuli responsive DNA walking device. Chemical Communications, 47(5):1428, 2011.
- [9] Universal Computing by DNA origami robots in a living animal, Yaniv Amir, Eldad Ben-Ishay, Daniel Levner, Shmulik Ittah, Almogit Abu-Horowitz and Ido Bachelet
- [10] <http://research.microsoft.com/en-us/projects/dna/>
- [11] <https://books.google.fi/books?id=kIMJg7tXPIC&pg=PA70&lpg=PA70&dq=dna+catalytic+gate&source=bl&ots=4bTBZeCLHU&sig=d2x4J7GPBvRyCSGQyHpYO7OQIO8&hl=en&sa=X&ei=AqsiVYWMN8ivsQGX8oLgCA&ved=0CB8Q6AEwAA#v=onepage&q=dna%20catalytic%20gate&f=false>; A Simple DNA gate Motif for synthesizing Large-Scale Circuits, Lulu Qian and Erik Winfree