

ارایه‌ی الگوریتمی جهت کاهش خطا و تعداد رشته‌ها در محاسبات دی.ان.ای

محمدرضا میبدی

دانشکده مهندسی کامپیوتر و فناوری اطلاعات،
دانشگاه صنعتی امیر کبیر

meybodi@ce.aut.ac.ir

جواد اکبری ترکستانی

دانشکده مهندسی کامپیوتر، دانشگاه آزاد اسلامی
اراک

j_akbari@iau-arak.ac.ir

تیمور ایزدی

دانشکده مهندسی کامپیوتر، دانشگاه آزاد اسلامی
اراک

t_izadi@iau-arak.ac.ir

چکیده

در محاسبات دی.ان.ای مهمترین عاملی که باعث محدودیت این روش و ایراد به آن شده است، بروز خطا و قابلیت اطمینان بسیار کم آن می باشد. خطا در کامپیوترهای دی.ان.ای در فاکتورهای زیادی رخ می دهد. در سال 1995، نپلان و همکارانش آزمایش آدلمن را انجام دادند. آن ها اعلام کردند ما همه مراحل آزمایش آدلمن را انجام دادیم، اما نتوانستیم نتایج نهایی غیر مبهم به دست آوریم. بعد ها دیگران نیز همین نظر را داشتند [10]. یکی دیگر از نقایص، کندی روند محاسبات می باشد. به گفته آدلمن زمان مورد نیاز برای تمام محاسبات با افزایش اندازه گراف به طور خطی رشد می کند. سوالی که به ذهن می رسد این است که در روند انجام محاسبات چه عواملی باعث بروز خطا می شود؟ آیا این عوامل غیر قابل کنترل هستند؟ در این تحقیق یکی از عواملی که باعث بروز خطا می شود شناسایی شده و الگوریتمی بسیار قوی برای کاستن و کنترل دقیق آن ارایه شده است. همچنین این روش باعث کاهش تعداد رشته های دی.ان.ای و افزایش مقیاس پذیری می شود

کلید واژه ها: محاسبات دی.ان.ای¹، مساله مسیر هامیلتونی، مسایل سخت محاسباتی²

1 مقدمه

رشته های دی.ان.ای و آر.ان.آ، که در طبیعت به وفور یافت می شود، توانایی ذخیره اطلاعات در حجم عظیمی را دارند [2]. همچنین با استفاده از دی.ان.ای می توان محاسباتی را انجام داد که انجام این محاسبات با استفاده از کامپیوترهای سلیکونی نیاز به صرف هزینه ای سنگین و حجم بالایی از پردازنده دارد [7], [5], [1]. اولین کسی که این ایده را ارائه کرد آقای لئونارد ادلمن از دانشگاه کالیفرنیا جنوبی بود. وی یکی از پیچیده ترین مسائل محاسبات (مساله فروشنده دوره گرد، مساله مسیر هامیلتونی) را با استفاده از دی.ان.ای حل کرد [1]. با اینکه بیش از ده سال از تولد محاسبات دی.ان.ای نمی گذرد امروزه یکی از شاخه های پیشرفت در علم محاسبات به حساب می آید. مزایا و معایب زیادی در محاسبه با دی.ان.ای وجود دارد اما همین قدر باید بدانیم که سرعت حل بعضی مسائل سخت محاسباتی، با صرف هزینه ای خیلی ارزانتر،

نسبت به سریعترین سوپر کامپیوترها بیشتر است. یک مزیت خیلی برتر در محاسبه با دی.ان.ای توانایی فوق العاده آن در ذخیره اطلاعات است. یک سانتی متر مربع از دی.ان.ای می تواند یک میلیون ترا بیت اطلاعات را ذخیره کند. بزرگترین تراشه های حافظه، امروزه می توانند کمتر از 20 ترابایت ذخیره کنند. با یک حساب سرانگشتی 50 میلیون کپی از این تراشه ها می توانند در یک سانتی متر مکعب از دی.ان.ای ذخیره شوند [14], [9], [6], [4].

2 یک بررسی در روند محاسبات دی.ان.ای نشان می دهد، که تمامی روشهای به کار رفته در حل مسائل سخت محاسباتی، از یک اسلوب پیروی کرده اند. آن، نگاشت این مسائل بر روی گرافها و درختها می باشد. در واقع ذات بیشتر این مسائل از گراف می باشد. اگر بخواهیم محاسبات دی.ان.ای را با این دید گاه بیان کنیم، می توانیم بگوییم این پردازش موازی در نودهای گراف، شکل می گیرد. این وظیفه را زوج های واتسون کریک مولکول های دی.ان.ای انجام می دهند. این موضوع بسیار پراهمیتی است، با توجه به آن می توان به یکی از عوامل مهم خطا خیز در این محاسبات دست یافت. در این مقاله نشان داده شده است که مکان خطا خیز همین نود ها هستند. آقای آدلمن در آزمایشش پیشنهاد داد که تعداد یال های گراف را به طور میلیونی افزایش دهیم تا خطا کاهش یابد. در شبیه سازی نشان داده شده که این روش بسیار موثر است، اما سربار میلیونی نودها باعث بروز کندی محاسبات، افزایش بی رویه حجم مساله، و افزایش قیمت و هم چنین از بین رفتن مقیاس پذیری خواهد شد. این مسئله باعث محدودیت محاسبات دی.ان.ای می شود. در این مقاله روشی ارائه گردیده است که در طی آن، با توجه به نوع مساله، افزایش هوشمندانه ای در تعداد یال ها خواهیم داشت. و نشان داده شده است که این افزایش هوشمندانه قابلیت اطمینان مطلوب را برای ما فراهم می کند. ضمن اینکه تعداد تکرار رشته های دی.ان.ای را در برخی موارد به طور میلیونی کاهش می دهد، همچنین ثابت شده است که احتمال تشکیل مسیر را افزایش خواهد داد. در یک نود تعداد یال های ورودی به نود را P و تعداد یال های خروجی از نود q نامیده شده است. برای نمونه اگر نودی داشته باشیم که دارای سه ورودی و دو خروجی باشد، در واقع $P=3$ و $q=2$. همچنین احتمال تشکیل یک مسیر خاص از بین مسیرهای تشکیل شونده، احتمال تشکیل یک مسیر نامیده شده است. و نسبت حداکثر مسیرهایی که تشکیل خواهد شد به کل مسیرهایی که در آن نود می تواند تشکیل شود، درصد مسیرهای تشکیل شونده نامیده شده است. خطای محاسباتی به خطایی که به لحاظ شکل مساله، قبل از تبدیل گراف مساله به رشته های دی.ان.ای ایجاد می شود، اطلاق می گردد. در روند تحقیق این نکته استنباط می شود، که احتمال تشکیل یک مسیر خاص

¹ DNA Computing

² NP_complete

1		1		1	1
1/2		1/2		1/2	1/2
1/3		1/3		1/3	1/3
1/4		1/4		1/4	1/4
1/5		1/5		1/5	1/5
1/q		1/q		1/q	1/q

(P = 1, q =)

2:

باشد افزایش P ها باعث افزایش احتمال تشکیل مسیر خواهد شد. (q ثابت و P متغیر). از موارد مطرح شده در بالا می توان به این نتیجه دست یافت که تنها زمانی افزایش P و یا q (یک کدام از آنها و نه هر دو) اثر بخش خواهد بود که مقدار آن مینیمم باشد. یعنی $\min(P, q)$ را باید افزایش داد (شکل 3). حال باید به این سوال پاسخ دهیم که تا چه حد افزایش دهیم؟ نتایج شبیه سازی نشان می دهد که ما تا حدی می توانیم افزایش دهیم که رابطه زیر برقرار باشد:

$$(2-2) (n+s)\min(P, q) \leq \max(P, q)$$

S مقدار افزایش یافته و n تعداد تکرار می باشد. حال این سوال مطرح می شود که اگر بخواهیم با افزایش توانان P ها و q ها، به احتمالی قابل ملاحظه و راه حلی دقیق برسیم چه راه حلی مناسب است؟

به تعداد ورودی و خروجی بستگی دارد. اگر به خواهیم دقیق تر بررسی کنیم در محاسبات دی.ان.ای، مسیرهای تشکیل شده نقش تعیین کننده دارند. این مسیرها در واقع از تک تک نودها عبور می کنند و احتمال تشکیل یک مسیر خاص به تک تک نودها بستگی دارد. همچنین احتمال تشکیل یک مسیر خاص در یک نود به تعداد ورودی و تعداد خروجی آن نود بستگی دارد. همچنین این احتمال دقیقاً برابر با درصد مسیر های تشکیل شونده در یک نود می باشد. البته زمانی صادق است که تعداد تکرار یال های ورودی و یال های خروجی برابر با یک باشد. این موضوع در شکل 2 نشان داده شده است. در شکل 2 نود های با ورودی و خروجی مختلفی نشان داده شده است. وقتی که تعداد تکرار یال ها برابر یک است، احتمال تشکیل یک مسیر خاص برابر با درصد تشکیل مسیرها است. نتایج تحقیق و شبیه سازی نشان می دهد، که اگر نودی با هر ورودی و هر خروجی در نظر گرفته شود، احتمال تشکیل مسیری خاص برابر با درصد مسیرهای تشکیل شونده است. این را با رابطه زیر نشان داده ایم. البته با تعداد تکرار هر یال به اندازه یک:

$$(1-2) \text{ احتمال تشکیل یک مسیر} = \frac{\min(P, q)}{(P * q)}$$

تمام مسیر هایی که ممکن است تشکیل شود برابر است با $(P * q)$. که حداکثر مسیرهایی که می تواند تشکیل شود برابر با مینیمم P و q می باشد، یعنی $\min(P, q)$. نتایج شبیه سازی در شکل 7 نشان داده شده است. می خواهیم با افزایش هوشمندانه P و q به افزایش صورت فرمول پردازیم، تا بتوانیم احتمال تشکیل یک مسیر را افزایش دهیم. در روند تحقیق به این نتیجه رسیدیم که مقدار افزایش هر ورودی را با توجه به q و تعداد هر خروجی را با توجه به P به همین نسبت افزایش دهیم. نتایج شبیه سازی نشان می دهد که این افزایش نسبتاً همیشه کارساز نیست. و در برخی موارد هیچ تاثیری در احتمال تشکیل یک مسیر ندارد. اما در برخی موارد احتمال تشکیل یک مسیر به طور چشمگیری افزایش خواهد یافت. از تحلیلی که در شکل بالا آورده شده است می توان به این نتیجه دست یافت که افزایش q تا زمانی این احتمال را بهبود می دهد که کمتر از P باشد. همین استدلال را برای افزایش P نیز خواهیم داشت. یعنی تا زمانی افزایش P کارساز خواهد بود که کمتر از q باشد. این موضوع را به دو دسته تقسیم می کنیم، ابتدا P را ثابت می گیریم و q را تغییر می دهیم. سپس q را ثابت می گیریم و P را تغییر می دهیم. به نتایج زیر دست پیدا کردیم: الف- هر چه اختلاف P و q زیادتر باشد، با P برابر کردن q ها، احتمال تشکیل مسیر افزایش می یابد. (P ثابت و q متغیر) ب- هر چه q بزرگتر از P

$$(9-2) = \frac{A}{B} = \frac{\min(p, q)}{p * q}$$

بنابراین رابطه احتمال تشکیل یک مسیر به دست می آید

فرضیات 2:

1) احتمال تشکیل یک مسیر در روش قبل:

$$f_k = \min(\text{ورودی}, \text{خروجی}) / (\text{ورودی} * \text{خروجی})$$

$$= \min(p, q) / (p * q)$$

$$P > q \quad (2)$$

3) احتمال تشکیل یک مسیر در روش ارایه شده

4) مقدار اضافه شده به خروجی r :

$$i = p \quad (5)$$

$$o = q + r \quad (6)$$

$$f_{k+1} = \min(p, q+r) / (p+q) \quad (7)$$

$$\sum_{i=1}^q w_i = k = (p - q) \% q \quad (8)$$

9) طبق تعریف باقیمانده:

$$k = (p - q) \% q = (p - q) - ((p - q) / q) * q$$

$$f_{k+1} \geq f_k \quad \text{حکم}$$

اثبات:

$$\sum_{i=1}^q r_i : r = \text{طبق تعریف}$$

$$, k = ((p - q) \% q) \begin{cases} 0 \\ 1 \text{ if } \sum_{r=0}^q w_r < k \end{cases} + w_i, w_i = \frac{p - q}{q} : r_i =$$

$$= \sum_{i=1}^q \left(\frac{p - q}{q} + w_i \right) = \sum_{i=1}^q \frac{p - q}{q} + w_i \quad r = \Rightarrow$$

$$\left(\sum_{i=1}^q \frac{p - q}{q} + \sum_{i=1}^q w_i \right) \quad \text{طبق فرض 8)}$$

$$= \left(\frac{p - q}{q} \right) q + (p - q) \% q \quad r =$$

$$\left(\left(\frac{p - q}{q} \right) q + (p - q) \right) \% q \quad \text{طبق فرض 9)}$$

$$r = p - q \quad (10)$$

$$i = p$$

رابطه پیشنهادی برای کلیه مسایل مبتنی بر محاسبات دی.ان.ای

برای اینکه بهترین درصد موفقیت را داشته باشیم طبق روال زیر عمل

می کنیم.

1) اگر $P > q$ آن گاه تمامی خروجی ها را به اندازه r_i اضافه می کنیم:

$$r_i = \frac{(P - q) * n}{q} + \omega_i$$

که ω_i بستگی به شماره هر نود خروجی دارد.

$$\omega_i = \begin{cases} 0 & \text{otherwise} \\ 1 & \text{if } \sum_{r=0}^k \omega_r < k \end{cases} \quad (4-2)$$

$$k = ((P - q) * n) \% q \quad (5-2)$$

در رابطه بالا n تعداد تکرار دلخواه عمومی می باشد

2) و اگر $P < q$ ، آن گاه هر ورودی را به اندازه s_i اضافه می کنیم

$$s_i = \frac{(P - q) * n}{P} + \omega_i \quad (6-2)$$

$$\omega_i = \begin{cases} 0 & \text{otherwise} \\ 1 & \text{if } \sum_{r=0}^k \omega_r < X \end{cases} \quad (7-2)$$

$$X = ((P - q) * n) \% P \quad (8-2)$$

اثبات:

با اثبات نشان داده ایم که در رابطه پیشنهادی احتمال تشکیل مسیر، بیشتر از حالت قبل می باشد. برای اثبات، یک سری فرضیات ارایه شده، و به کمک این فرضیات، حکم مساله اثبات شده است

فرضیات 1:

• اصل ضرب: فرض کنید که یک کار را بتوان با دو عمل پایایی A و B انجام داد. اگر عمل A به m طریق و به دنبال آن عمل B بتواند به n طریق انجام پذیرد. آن گاه این کار به mn طریق انجام می پذیرد.

اصل لانه کبوتر: اگر n کبوتر و $n-1$ لانه کبوتر داشته باشیم، اگر به هر کبوتر یک لانه برسد، آن گاه قطعاً یک کبوتر بدون لانه خواهد ماند

به کمک فرضیات بالا می توان احتمال تشکیل یک مسیر را به دست آورد.

(طبق اصل شمارش) $B = p * q =$ تعداد کل مسیر های متفاوتی که می تواند شکل بگیرد

(طبق اصل لانه کبوتر) $A = \min(p, q) =$ تعداد مسیر هایی که در حالت فعال شکل می گیرد

باشد به دست می دهند. با وجود این وقتی با گرافهای مشخصی برخورد می کنیم اغلب به آزمون و خطا همراه با تعدادی ملاحظات یاری دهنده متوسل می شویم [8],[11],[3],[13],[12].

5) الگوریتم پیشنهادی برای مساله مسیر هامیلتونی

Input :Graph G,nodes

Step 0) Represent nodes,edges,paths with DNA

Step 1) For each node we increase inputs by r_i and outputs by s_i that :

$$\frac{(P-q)*n}{q} + \omega_i r_i =$$

$$\omega_i = \begin{cases} 0 & \text{otherwise} \\ 1 & \text{if } \sum_{r=0}^i \omega_r < k \end{cases}$$

$$k = ((P-q)*n)\%q$$

$$\frac{(P-q)*n}{P} + \omega_i s_i =$$

$$\omega_i = \begin{cases} 0 & \text{otherwise} \\ 1 & \text{if } \sum_{r=0}^i \omega_r < X \end{cases}$$

$$X = ((P-q)*n)\%P$$

Step 2) Fill tubes with all possible paths

Step 3) Select paths

Step 4) Select paths of correct length

Step 5) Select paths without duplicate vertexes

Step 6) If everything remains

Then return "yes"

Else return "no"

در الگوریتم پیشنهادی همانطور که آورده شده است، نسبت به الگوریتم های رایج یک مرحله اضافه گردیده است، که در مرحله 1 آورده شده است. در واقع در مرحله 1 در هنگام تبدیل گراف به رشته های دی.ان.ای با توجه به روش ارائه شده، هر رشته را به تعداد مشخص تولید می کنیم. و بقیه حالات شبیه مراحل قبل می باشد.

$$) 10 \text{طبق } o = q+r = q + (p-q) = p ($$

$$(11) \Rightarrow i = o = p$$

$$\Rightarrow \min(i, o) = p$$

$$\frac{\min(i, o)}{p * q} = \frac{p}{p * q} \quad f_{k+1} = \Rightarrow$$

طبق فرض، $p > q$

$$\Rightarrow \frac{1}{p * q} > 0 \quad p > 0, q > 0$$

$$\Rightarrow p * \left(\frac{1}{p * q}\right) > \frac{q * 1}{p * q}$$

$$\Rightarrow f_{k+1} > f_k \text{ وحکم ثابت می شود}$$

4 مساله مسیر هامیلتونی

در سال 1859 سر ویلیام رون همیلتون (1805-1865) ریاضیدان ایرلندی بازی ای ابداع کرد که آن را به یک کارخانه سازنده اسباب بازی در دوبلین فروخت. این بازی یک دوازده وجهی منتظم چوبی با 20 نقطه گوشه ای (20 راس) بود و هر یک از این راسها نام یکی از شهرهای مهم را داشت. بازی عبارت بود از یافتن دوری در امتداد یالهای این دوازده وجهی به طور که هر یک از شهرها (دقیقا یک بار) روی این دور قرار گیرد. شکل زیر گراف مسطح این جسم افلاطونی اند. یکی از دورهای مورد نظر را با یال های سیاه مشخص کرده ایم. این توضیح ما را به تعریف زیر هدایت می کند: اگر $G(V,E)$ گراف یا گراف چند گانه ای باشد که در آن $V \geq 3$ ، می گویند G دور همیلتونی دارد هر گاه در G دوری حاوی هر یک از راس های V وجود داشته باشد. یک مسیر هامیلتونی مسیری (و نه دوری) است که حاوی هر یک از راسها باشد. اگر گرافی حاوی دوری همیلتونی داشته باشیم می بینیم که از حذف هر یک از یال های این دور یک مسیر همیلتونی به دست می آید. ولی این امکان نیز وجود دارد که گرافی حاوی مسیری همیلتونی باشد بی اینکه دوری همیلتونی داشته باشد. چنین به نظر می رسد که وجود دوری (مسیری) همیلتونی و وجود مدار (پیگردی) برای گرافها مسایل مشابهی باشند. دور (مسیر) همیلتونی برای این هدف طرح شده است که از هر راس متعلق به گراف فقط یک بار بگذرد. مدار اوپلری گراف را چنین طی می کند که هر یک از یالها فقط یکبار پیموده می شود. متاسفانه هیچ پیوند یاری دهنده ای بین این دوایر وجود ندارد و بر خلاف وضعیت مربوط به مدارهای اوپلری (و پیگرد های اوپلری) شرایط لازم و کافی برای تضمین وجود دور همیلتونی (یا مسیر همیلتونی) در گرافی مانند G در دست نیست. اگر گرافی یک دور همیلتونی داشته باشد در این صورت این گراف دست کم همبند است. قضیه های زیادی وجود دارند که شرایط لازم کافی برای آن که یک گراف همبند دور یا مسیر همیلتونی داشته

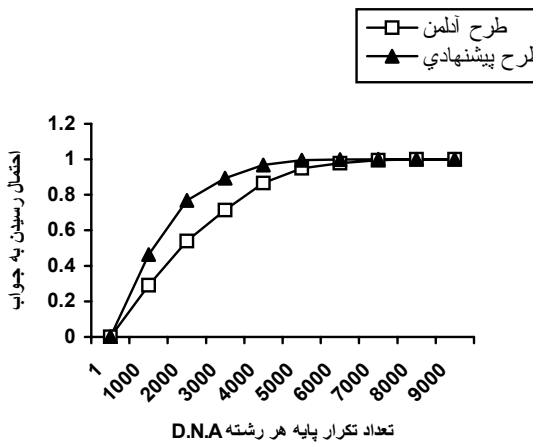
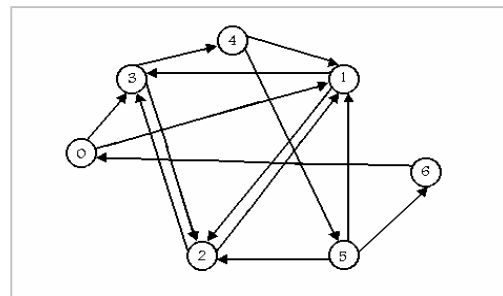
5 شبیه سازی و ارزیابی کارایی

در این مقاله الگوریتم های قبلی و الگوریتم ارایه شده، شبیه سازی شده است. سپس نشان داده شده این الگوریتم با تعداد تکرار های مساوی نسبت به الگوریتم قبلی، درصد احتمال تشکیل مسیر در مساله مسیر هامیلتونی آدلمن را افزایش می دهد. در شکل 4 گراف ای که آقای آدلمن حل کرد آورده شده است در این مقاله گراف مساله گراف هامیلتونی بالگوریتم پیشنهادی و روش آدلمن شبیه سازی و مقایسه شده است. سپس میزان درصد موفقیت در نود های هر روش را به طور مجزا نشان داده ایم. در شکل 5، شماره. هر نود در سطر نمودار به کار رفته است. و در ستون نمودار میزان درصد موفقیت در هر دو روش نشان داده شده است. مشاهده می شود که در هر نود افزایش احتمال تشکیل مسیر را داریم. اما این فقط مقایسه ای در میان نودها می باشد، و باید مقایسه را در حالت کلی برای تمام حالات بررسی کنیم. برای این کار شبیه سازی را برای مساله گراف هامیلتونی بالا کامل کردیم. نتایج شبیه سازی در هر دو الگوریتم ارائه شده و الگوریتم های قبلی در نمودار شکل 6 نشان داده شده است. نمودار شکل 6 که از شبیه سازی هر دو روش به وجود آمده، نشان دهنده بهبود احتمال به جواب رسیدن در روش پیشنهادی و همچنین کاهش تعداد تکرار یال ها می باشد. به این معنی که در روش پیشنهادی با صرف مقدار دی.ان.ای کمتر، به افزایش درصد احتمال دست پیدا کرده ایم. این خود باعث افزایش مقیاس پذیری در محاسبات دی.ان.ای خواهد شد

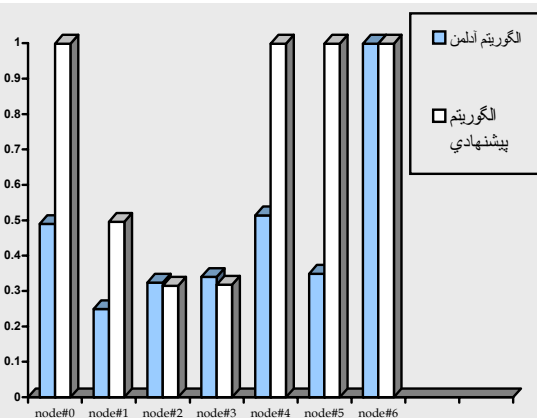
6 نتیجه گیری

در این مقاله نقطه خطا خیز در محاسبات دی.ان.ای شناسایی و معرفی شده است. والگوریتمی ارایه شده است که این خطا را تا حد بسیار زیادی کنترل می کند. درصد احتمال تشکیل مسیر مورد نظر، افزایش یافته است. مقدار رشته های دی.ان.ای به کار رفته در محاسبات دی.ان.ای کاهش پیدا کرده است. به خاطر همین کاهش تعداد رشته ها، مقیاس پذیری در محاسبات در محاسبات دی.ان.ای (افزایش حجم مساله) نیز افزایش یافته است. عیب این روش پیش پردازش روی مساله می باشد، که به خاطر اضافه کردن مرحله 1 می باشد. هزینه این پردازش برابر با تعداد نودها می باشد. اگر تعداد این نود ها برابر K باشد، این هزینه برابر $O(K)$ می باشد. ما این الگوریتم را روی مساله مسیر هامیلتونی پیاده سازی کردیم و نتایج را ارزیابی نمودیم.

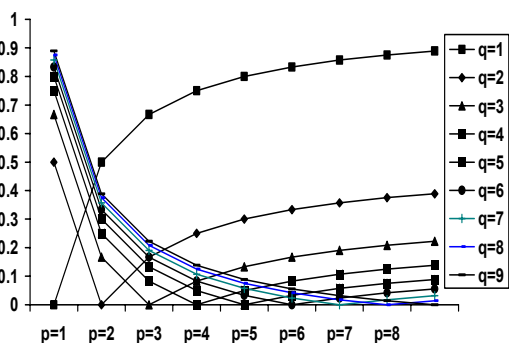
شکل 4: گراف مساله مسیر هامیلتونی که آدلمن با محاسبات دی.ان.ای حل کرد



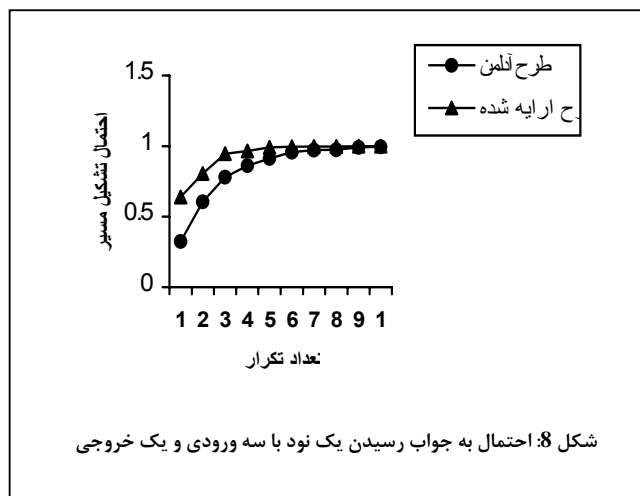
شکل 6: احتمال به جواب رسیدن در مدل پیشنهادی و مدل آدلمن



شکل 5: درصد موفقیت (تشکیل مسیر)، در هر نود به طور مجزا نشان داده شده است



شکل 7: احتمال رسیدن به جواب در نودهای باورودی P و خروجی q



7 مراجع:

- [1] Leonard Adleman, "Molecular computation of solutions to combinatorial problems", Science, 1994
- [2] Roger L.P. Adams, John T. Knowler, David P. Leader, "The biochemistry of Nucleic Acids", Chapman and Hall, tenth edition, 1986.
- [3] A.V. Aho, J.E. Hopcroft, L.D. Ulman, "The design and analysis of computer algorithms", Addison-wesley, 1994.
- [4] Martyn Amos, the Ph.d thesis, "DNA computation", WarWick university, 1997.
- [5] Eric Bach, Anne Condon, Elton Glaser, Celena Tanguay, "DNA Models and algorithms for Np-complement problems", IEEE computer society press, 1996.
- [6] Tiina ZINGEL, "Formal models of DNA computing :A survey", 24 May 1999
- [7] Lipton, Richard J. and Baum, Eric B., Eds., "DNA Based Computers: Proceedings of a DIMACS workshop", Princeton University, 1996
- [8] Morimoto, Nobuhiko, Masanori Arita and Akira Suyama, "Solid phase DNA solution to the Hamiltonian path problem..", Rubin and Wood, 1997
- [9] Martyn Amos, Alan Gibbons, and Paul E. Dunne. "The complexity and viability of DNA computations.: Proceedings of the First International Conference on Bio-Computing and Emergent Computation", University of Skovde, Sweden, 1997
- [10] Martyn Amos, Alan Gibbons, and David Hodgson, "Error-resistant implementation of DNA computations." : In Proceeding of the second Annual Meeting on DNA Based Computers.
- [11] Eric Bach, Anne Condon, Elton Glaser, and Celena Tanguay. "DNA Models and Algorithms for NP-complete Problems.", IEEE Computer Society Press, 1996
- [12] R.E. Neapolitan and K. Naimipour, "Foundations of Algorithms Using C++ Pseudo Code, Second edition", Jones and Barlett publishers, 1998
- [13] Mahmoud Naghibzadeh, "Computer Algorithms Analysis, Design, and Performance Evaluation", Astane ghodse razavi Publishers, 2000
- [14] Zucca M., "DNA based computational models", phd Thesis: Politecnico Di Torino, 2000