

# بهینه سازی تخصیص زیر پرس وجوها در پایگاه داده توزیعی با استفاده از الگوریتم فرهنگی

ناصر سیه چهره<sup>۱</sup> ، بهاره منورشرافیان<sup>۲</sup>

<sup>۱</sup> دانشگاه آزاد اسلامی واحد قزوین ، دانشکده ی مهندسی برق ، کامپیوتر و فناوری اطلاعات ، [Nasser\\_s1@yahoo.com](mailto:Nasser_s1@yahoo.com)

<sup>۲</sup> دانشگاه آزاد اسلامی واحد قزوین ، دانشکده ی مهندسی برق ، کامپیوتر و فناوری اطلاعات ، [sharafian\\_b@yahoo.com](mailto:sharafian_b@yahoo.com)

چکیده - پردازش پرس و جو مفهوم مهمی در حوزه ی پایگاه داده ی توزیع شده می باشد [۱] که در همین راستا پردازش پرس و جو ی توزیع شده با توجه به اینکه هدف کم کردن یا به حداقل رساندن هزینه ی عملیاتی می باشد، فاکتور مهمی در کل کارایی سیستم پایگاه داده ی توزیع شده می باشد.

ما در این مقاله از معماری مشتری /خدمتگذار استفاده می کنیم ، این نوع معماری بعد از معماری متمرکز و به علت نا کارآمدی معماری متمرکز در سیستم های بزرگتر بوجود آمد. در این نوع معماری قسمت های مختلف پایگاه داده در کامپیوتر های مختلف قرار می گیرند. در این روش یک کامپیوتر نقش سرور را به عهده می گیرند و کاربران می توانند با استفاده از کامپیوتر های دیگر و متصل شدن به سرور از پایگاه داده استفاده کنند. همانطور که مشخص است کاربران و کامپیوترها برای اتصال به سرور نیازمند به شبکه کامپیوتری هستند پس در واقع شبکه های کامپیوتری یکی از ملزومات این نوع معماری به حساب می آید.

کلید واژه - بهینه سازی، الگوریتم بهینه سازی زیر پرس و جو، الگوریتم های قطعی -الگوریتم فرهنگی - جهش

می شود. درخواست های کاربران به صورت دستورات RPC و دستورات SQL ارسال می شود. برای ارسال و دریافت نیاز به نرم افزار هایی مانند ODBC و JDBC هست که وظیفه ارتباط با سرور و ارسال پرس و جوها و دریافت نتایج را به عهده دارد. برای پیاده سازی سرورهای تراکنشی از تعدادی پردازش که حافظه مشترک دارند استفاده می شود که معمولا برای بهبود پردازش ها هر کدام از آنها Multithread مورد استفاده قرار می گیرند.

برای بهینه سازی پرس و جو در یک محیط کلاینت/سرور باید اطلاعات کافی در مورد مکان های داده ها داشته باشیم تا بتوانیم با توجه به آن بهترین روش را برای دسترسی به داده ها پیدا کنیم.

مهمترین مسئله در اینجا تجزیه کردن پرس و جو ها می باشد. پردازش پرس و جوی توزیع شده در سه مرحله ی فاز پردازش محلی، فاز ساده سازی و فاز پردازش انجام می گیرد، در فاز پردازش محلی گزینش ها و انتخاب انجام می گیرد و در فاز ساده سازی از یک کهنده برای کم کردن اندازه ی رابطه ها استفاده می شود و در فاز پردازش نهایی نتایج رابطه ها در یک جا جمع شده و نتیجه ی نهایی پرس و جو ایجاد و فرستاده می شود. در اینجا به این خاطر که از موازی سازی کمتری استفاده می شود پدیده ی سربرار زیاد انتقال را داریم [۲].

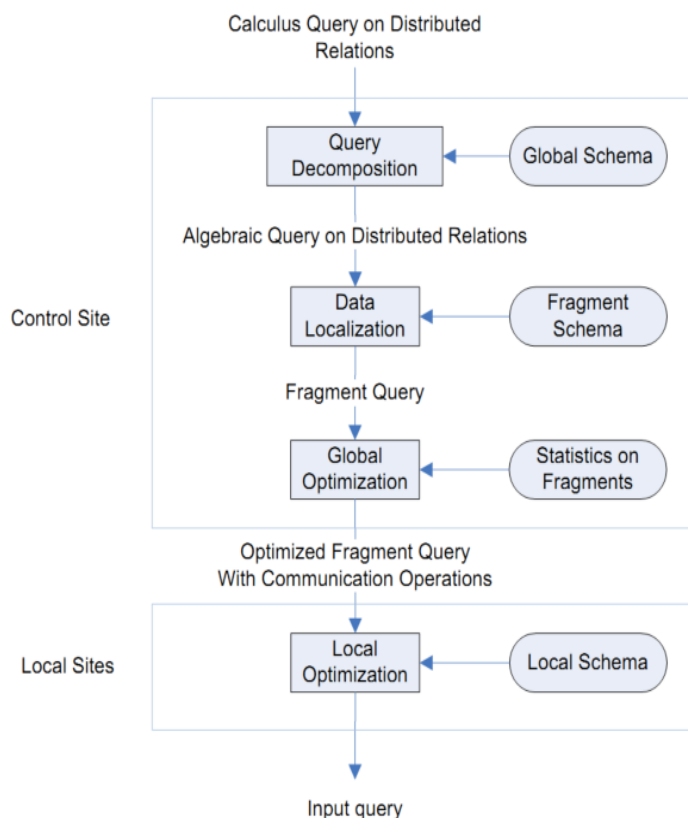
در پردازش پرس و جو ی توزیع شده، یکسری عملیات تحت عنوان عملیات عمومی نیز وجود دارد که این

پایگاه داده ها در این نوع معماری بر حسب کاربری به دو بخش back-end و Front-end تقسیم می شود، Back-end وظیفه بررسی و کنترل دسترسی ها، بررسی و بهینه سازی پرس و جوها و کنترل همزمانی ها و سالم بودن پایگاه داده ها را به عهده دارد، Front-end شامل ابزار هایی برای نمایش و زیباسازی نتایج پرس و جو ها مثل ابزارهای تولید فرم ها و ابزار های گزارش گیری می باشد. برای ایجاد ارتباط درست میان دو قسمت فوق نیازمند یک لایه و بستر می باشیم. این بستر می تواند از دو طریق دستورات SQL و یا API ها برقرار شود.

در مقایسه این نوع معماری با معماری متمرکز با استفاده از Mainframe ها می توان افزایش میزان کاربری سیستم با توجه به هزینه، آسان تر شدن گسترش و توزیع منابع، تولید واسط های کاربر بهتر و راحت تر شدن نگهداری سیستم را نام برد.

در این نوع معماری، سرورها از لحاظ عملکردی به دو بخش تقسیم می شوند، یکی سرورهای داده ای که این نوع سرور ها بیشتر در سیستم های شی گرا مورد استفاده قرار می گیرند و دیگری سرور های تراکنشی که این نوع سرور ها بیشتر در سیستم های رابطه ای مورد استفاده قرار می گیرند.

در سرورهای تراکنشی که به آنها سرورهای پرس و جو هم گفته می شود روش کار بدین صورت است که درخواست های کاربران به این سیستم ها ارسال می شود و نتیجه در این سرور ها تولید و به کاربر ارسال



شکل ۱: بهینه سازی پرس و جوی توزیعی

عملیات شامل قسمت بندی یک رابطه در داخل قطعه ها، اجتماع قطعه ها از کل رابطه و انتقال یک رابطه / قطعه از یک پایگاه داده به دیگر پایگاه داده می باشد [۳].

برای ساده سازی انتقال داده و برای مرحله ی پردازش پرس و جوی توزیع شده از نیم پیوند و پیوند متوالی استفاده می شود، در نیم پیوند عملیات پیوند برای کاهش پردازش پرس و جوی استفاده می شوند و با استفاده از احیا کننده ی پیوند یک پرس و جوی در داخل پیوند های متوالی ترجمه می شود و هر پیوند بوسیله ی یک عملگر رابطه ای به طور محلی با توجه به محل عملگر های دیگر پیاده سازی می شود. همچنین می توان از ترکیب پیوند ها و نیم پیوند ها برای بهبود پردازش پرس و جوی توزیع شده استفاده کرد [۴]، [۵].

## ۲- جزئیات مدل های هزینه ی پردازش پرس و جوی

Valduriez [۶] یک متدلوژی پردازش پرس و جوی

توزیع شده شرح داده است که در شکل ۱ نشان داده شده است.

ورودی در شکل فوق یک پرس و جوی بر روی داده توزیع شده در حساب رابطه ای می باشد. لایه های اصلی برای پرس و جوی توزیع شده در داخل یک توالی عملیات محلی بهینه شده قرار گرفته است، هر عامل بر روی یک پایگاه داده محلی است. این لایه ها تجزیه توابع پرس و جوی، محلی کردن داده، بهینه سازی سراسری پرس و جوی و بهینه سازی محلی پرس و جوی را اجرا می کنند. تجزیه محلی کردن پرس و جوی داده مربوط به پرس و جوی بازنویسی می باشد. سه لایه اول بوسیله یک محل مرکزی انجام می شود و مورد استفاده اطلاعات سراسری است.

## ۲-۱- محاسبه ی پردازش محلی

برای محاسبه پردازش محلی، بستگی به نوع عملیات، سرعت cpu، اندازه ی رابطه و سرعت ورودی سایت منتخب می باشد. پردازش محلی از دو بخش ورودی/خروجی و پردازش تشکیل شده است که به صورت زیر محاسبه می شود.

مجموع تمام زیر پرس و جو ها ی سایت  $t =$

پردازش محلی (ورودی/خروجی) (تعداد بلاک رابطه ی  $i$  که توسط زیرپرس وجوی  $j$  رابطه ی  $K$ ) + (رابطه یا رابطه ها  $i$  توسط زیرپرس وجوی  $j$  برای پرس وجوی  $k$ )

مجموع تمام زیر پرس و جو ها ی سایت  $t =$

پردازش محلی (cpu) (تعداد بلاک رابطه ی  $i$  که توسط زیرپرس وجوی  $j$  رابطه ی  $K$ ) + (رابطه یا رابطه ها  $i$  توسط زیرپرس وجوی  $j$  برای پرس وجوی  $k$ )

## ۲-۲- محاسبه ی هزینه ی ارتباطی

برای محاسبه ی هزینه ارتباطی فرض می کنیم گرافی مانند شکل زیر داریم که در واقع موقعی هیچ رابطه ای برای اتصال وجود نداشته باشد در هر کدام از سایت ها عملیات اتصال به گره بالایی صورت می گیرد

بهینه سازی محلی توسط مکان های محلی انجام شده است.

سه نوع از انواع معمول الگوریتم ها برای بهینه سازی Join-ordering الگوریتم های قطعی ، ژنتیک و تصادفی هستند [۷].

زمان پردازش محلی شامل موارد زیر می باشد: نوع عملیات، اندازه ی رابطه ورودی ، سرعت CPU و سرعت ورودی و خروجی سایت منتخب پس ،

(۱)

$$\text{Total cost} = \sum_j LocalP_j^K + COMM_j^K$$

که  $j$  نشان دهنده ی تعداد زیر پرس وجو ها و  $k$  نام خود پرس وجوی افراز شده می باشد.  $LocalP_j^K$  نشان دهنده ی زمان پردازش محلی  $j$  زیر پرس و جو برای پرس وجوی  $k$  می باشد.  $COMM_j^K$  یعنی زمان انتقال رابطه مورد نظر که در واقع پرس و جو  $k$  به  $j$  زیر پرس و جو افراز شده است.

$$subQ_{jt}^k = \begin{cases} \text{صفر عدم وجود زیرپرس و جو ی } j \text{ در سایت } t \text{ برای پرس و جو ی } k \\ \text{یک وجود زیرپرس و جو ی } j \text{ در سایت } t \text{ برای پرس و جو ی } k \end{cases}$$

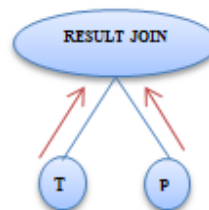
### ۳- زمان پاسخ

برای محاسبه ی زمان پاسخ در اینجا ، در سیستم پایگاه داده ی توزیعی ، تجزیه ی یک پرس و جو به یک زیر پرس و جو این امکان را می دهد که فرآیند پردازش به صورت موازی انجام شود و همچنین رابطه های میانی می توانند برای پردازش به سایت های مورد نیاز به در کل دونوع اجرای موازی صورت موازی فرستاده شوند. وجود دارد : یکی عملیات درونی و دیگری عملیات ، بروی intra-operation داخلی، در عملیات درونی عملیات انتقال انفرادی خط لوله به صورت موازی صورت می گیرد که هر کدام از سایت ها به صورت موازی کار می کنند. در عملیات داخلی، چندین زیر پرس و جو در یک پرس و جو می توانند به صورت موازی اجرا شوند. [۸]

= زمان پاسخ زیر پرس و جوی J توسط پرس و جوی k  
(پردازش محلی زیر پرس و جوی J توسط پرس و جوی k) + (هزینه ی ارتباطی زیر پرس و جوی J که توسط پرس و جوی k در برگ اول p) (زمان پاسخ زیر پرس و جوی J توسط پرس و جوی k) +

نکته :قسمت سوم رابطه فوق به صورت تابع بازگشتی می باشد یعنی از برگ شروع کرده تا به ریشه می رسد و زمان های پاسخ را جمع می کند.

در اینجا زمان های پاسخ مختلفی به وجود می آید و حالت های مختلفی باید بررسی شود اما با رشد درجه ی موازی سازی زمان پاسخ کمینه می شود و با کمینه شدن



شکل ۲: گراف با دوگره که هر نود معرف یک سایت می باشد.



شکل ۳: گراف با یک گره سمت چپ یا راست که هر نود معرف یک سایت می باشد.

به تعداد برگهای زیر درخت و جمع تمام زیر پرس و جوی های در سایت t و جمع تمام زیر پرس و جوی های در سایت P = هزینه ی ارتباطی قسمت اول (وجود زیر پرس و جوی J در سایت P برای برگ اول توسط پرس و جوی K) + (وجود زیر پرس و جوی J در سایت T برای برگ اول توسط پرس و جوی K) جمع همه رابطه و یا نتیجه میانی i = هزینه ی ارتباطی قسمت دوم + (انتقال نتیجه میانی i توسط زیر پرس و جوی J در برگ اول و یا دوم و یا هر دو در سایت p) (نتیجه i بوسیله ی زیر پرس و جوی J در برگ اول و یا برگ دوم سایت p، و یا هر دو توسط پرس و جوی k) اگر هر دو پیوند قبلی و اجرای آن در یک سایت باشد رابطه ی هزینه ارتباطی قسمت دوم صفر می شود.

```

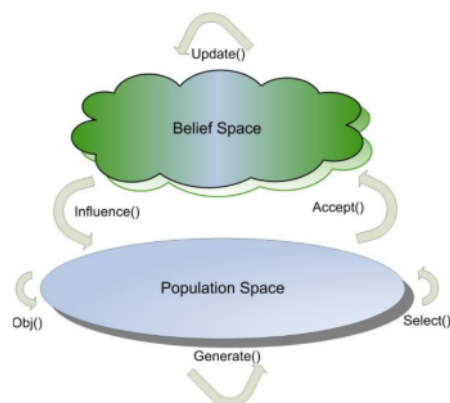
Update(BLF(t), Acceptance (POP(t)));
Variation(POP(t), Influence(BLF(t)));
t=t+۱;
Select POP(t) from POP(t-۱);
Until termination condition achieved;
End

```

زمان کل و با به کارگیری بهبود منابع ، بهره ی ورودی سیستم افزایش پیدا می کند.

#### ۴- الگوریتم فرهنگی

الگوریتم های فرهنگی توسط (Reynolds, ۱۹۹۴) معرفی شده است [۹]. که شامل دو مولفه فضای باور و فضای جمعیت می باشد. فضای جمعیت، شامل مجموعه ای منحصربفرد که کاندید جواب مسئله می باشد. فضای باور اطلاعات و تجربیات را ذخیره می کنند. که بوسیله منحصربفرد ها در طول پردازش تکاملی مورد استفاده قرار می گیرد [۱۰]. همانطور در شکل ۴ نشان داده شده، منحصربفرد ها براساس تابع کارایی ارزیابی می شوند. بعضی منحصربفرد ها از فضای جمعیت بوسیله تابع پذیرش انتخاب می شوند و اطلاعات فضای باور را بروز رسانی می کنند.



شکل ۴- چارچوب الگوریتم فرهنگی

همچنین شبه کد الگوریتم مانند زیر می باشد :

#### ۴-۱- شرح الگوریتم پیشنهادی

ابتدا فضای جمعیت بصورت تصادفی مقداردهی اولیه می شود. هریک از اعضای فضای جمعیت نشان دهنده ی یک راه حل برای مسئله می باشد. درواقع هریک از راه حل ها یک کروموزوم نامیده می شود. کروموزوم ها براساس شایستگی آنها ارزیابی می شوند. نمایش کروموزوم ها بصورت اعداد صحیح می باشد. تعداد اعضای فضای جمعیت به تعداد عملیات ها پرس و جو و و تعداد سایت ها می باشد. پس اندازه ی فضای جمعیت به تعداد سایت ها موجود در شبکه می باشد. عمل جهش در انتهای کار بصورت تصادفی روی یکی از ژن های کروموزوم صورت می گیرد. برای crossover کردن و تولید مثل از روش one-point بکار گرفته می شود. برای انتخاب کروموزوم ها برای عمل crossover و جهش از تابع Roulette Wheel selection استفاده می شود. در این الگوریتم به کروموزوم ها با شایستگی بالا توجه شده است به عبارت دیگر به نخبه گرایی توجه زیادی دارد. بخاطر اینکه الگوریتم زود به همگرایی برسد که باعث می شود زمان پاسخ زیرپرس وجو کمینه می شود. برای اینکار ما در این الگوریتم فشار انتخاب را زود کاهش داده ایم .

#### ۴-۲- طراحی فضای باور

ابتدا، همه کروموزوم ها در یک مجموعه خالی قرار می گیرد. ۲۰٪ از منحصربفرد های فضای باور ، بعنوان

```

Begin
t=۰;
Initialize Population Space POP(t);
Initialize Belief Space BLF(t);
Repeat
Evaluate Population POP(t);

```

روش را با دقت بیشتری انتخاب نمود تا به جواب با کارایی بالاتری دست پیدا کنیم.

## مراجع

1. Alom. B.M. Monjurul, Henskens.Frans And Hannaford.Michael, "Query Processing and Optimization in Distributed Database Systems", *IJCSNS International Journal of Computer Science and Network Security*, VOL.9 No.9, September 2009.
2. C. Wang and M.-S. Chen, "On the Complexity of Distributed Query Optimization," *IEEE vol. 8*, pp. 650-662, 1994.
3. C. Liu and C. Yu, "Performance Issues in Distributed Query Processing," *IEEE*, vol. 4:8, pp. 889-905, 1993.
4. M. S. Chen and P. S. Yu, "Combining Join and Semi Operations for Distributed Query Processing," *IEEE Trans. of Knowledge & Data Engineering*, vol. 5:3, pp. 534-542, 1993.
5. M. S. Chen and P. S. Yu, "A Graph Theoretical Approach to Determine a Join Reducer Sequences in Distributed Query Processing," *IEEE Trans. of Knowledge & Data Engineering*, vol. 6:1, pp. 152-165, 1994.
6. J M. Tamer Özsu, Patrick Valduriez, "Principles of Distributed Database systems, third Edition", Prentice Hall, ISBN 0-13-659707-6, 2011
7. Kristina Zelenay, "Query Optimization", ETH Zürich, Seminar Algorithmen für Datenbanksysteme, June 2005.
8. Cornell, D.W. & Yu, P.S. 1989. On optimal site assignment for relations in the distributed database environment. *IEEE Transactions on Software Engineering*, 15 (8), 1004-1009.
9. Reynolds, R. G., "An Introduction to Cultural Algorithms", *Proceedings of the Third Annual Conference on Evolutionary Programming*,

منحصربفردهای فضای جمعیت انتخاب می شوند. در واقع این 20٪ برای عمل crossover و جهش انتخاب می شوند.

## ۴-۳- طراحی تابع پذیرش

تابع پذیرش در واقع تصمیم گیری می کند که کدام منحصربفردها روی فضای باور تاثیر پیدا کنند. که در واقع درصدی از منحصربفردهای فضای جمعیت می باشند. یعنی منحصربفردهایی با شایستگی بالا جایگزین منحصربفردهایی با شایستگی پایین می شود.

## ۴-۴- طراحی تابع موثر

بعد از اینکه منحصربفردها بعد از  $k$  تا تولید نسل ارزیابی شده اند ، 15٪ از منحصربفردهایی از فضای باور جایگزین منحصربفردها با شایستگی پایین می شود.

## ۵- نتیجه گیری

در روش پیشنهادی برای حل مسئله نحوه تخصیص زیر پرس وجوها به سایت های مختلف در فضای شبکه می باشد. در واقع زمان پاسخ کمینه می شود. بعلاوه اینکه الگوریتم فرهنگی یک الگوریتم هیوریستیک می باشد امکان موازی سازی در سایت مختلف را دارا می باشد. و باعث می شود زیرپرس وجوهای بیشتری را به سایت ها داده و امکان صرفه جویی را بر آورده می سازد. و به نسبت روش های با شمارش جامع از کارایی و عملکرد بهتری برخوردار می باشد. ولی می توان برای بدست آوردن جواب بهینه تر می توان پارامترهای این

San Diego, California, pp. 131-139, February 24-26, 1994.

10. X. Yuan, Y. Yuan. Application of cultural algorithm to generation scheduling of hydrothermal systems. *Energy Conversion and Management*, vol. 47, no. 10-11, pp. 2192-2201, 2006.