انتخاب دید جهت ذخیره‌سازی دید در پایگاه داده تحلیلی با استفاده از الگوریتم فرهنگی ترکیبی

پروانه شایق بروجنی[۱]، دانشجوی کارشناسی ارشد؛ نگین دانشپور[۲]، استادیار

۱- دانشکده مهندسی کامپیوتر - دانشگاه تربیت دبیر شهید رجایی - تهران - ایران - p.shayegh@srttu.edu
۲- دانشکده مهندسی کامپیوتر - دانشگاه تربیت دبیر شهید رجایی - تهران - ایران - ndaneshpour@srttu.edu

چکیده: پایگاه داده تحلیلی حجم زیادی از داده‌ها که در سیستم‌های تصمیم‌گیرنده و گزارش‌گیر مورداستفاده قرار می‌گیرد را ذخیره می‌کند. در این سیستم‌ها سرعت پاسخ‌گویی به پرس‌وجوها به علت حجم زیاد داده‌های ذخیره‌شده، پایین است. ازآنجایی‌که این سیستم‌ها عموماً مورداستفاده مدیران ارشد در سازمان‌های مختلف هستند، درنتیجه افزایش سرعت در این سیستم‌ها حائز اهمیت است. یکی از روش‌های افزایش سرعت، ذخیره دیدها جهت پاسخ‌گویی به پرس‌وجوها است. از طرفی ذخیره کلیه دیدها نیاز به حجم حافظه زیاد دارد و غیرممکن است. راهکار، انتخاب یک مجموعه دید مناسب از بین همه دیدها است. مسئله جستجو و انتخاب یک زیرمجموعه از یک فضای بزرگ یک مسئله NP hard است. تاکنون الگوریتم‌های زیادی برای یافتن این مجموعه معرفی شده‌اند که دراین‌بین الگوریتم‌های تکاملی بسیار مورداستفاده قرار گرفته‌اند. در این مقاله از الگوریتم فرهنگی ترکیبی برای پیدا کردن N دید که بهترین نتیجه را داشته باشند استفاده می‌شود. آزمایش‌ها نشان می‌دهد که این الگوریتم در مقایسه با الگوریتم انتخاب دید ژنتیک، جستجوی فاخته و الگوریتم تفاضلی دارای هزینه کم‌تر و سرعت بیش‌تری است.

کلمات کلیدی: پایگاه داده تحلیلی، دید ذخیره‌شده، الگوریتم فرهنگی ترکیبی.

Materialized View Selection using Hybrid Cultural Search Algorithm

P. Shayegh Boroujeni[1], MSc Student; N. Daneshpour[2], Assistant Professor

1- Faculty of Computer Engineering, Shahid Rajaee Teacher Training University Tehran, Iran Email: p.shayegh@srttu.edu
2- Faculty of Computer Engineering, Shahid Rajaee Teacher Training University Tehran, Iran Email: ndaneshpour@srttu.edu

Abstract: A data warehouse stores a large amount of data, which are usually used in decision support systems. Response time of these systems is too high because of their huge data. Since these systems are generally used by organization's supervisors, reducing this response time is important. One of the major solutions for this problem is view materialization. Materialization of all views is impossible according to the constraint on memory space and the cost of maintenance these views. So, it is needed to select proper views to be materialized. Selection of these views is a kind of searching in a huge space that is considered as NP hard problem. Several methods are proposed to address this problem until now. Evolutionary algorithms are mostly used in solving MV problems. In this paper, Hybrid Cultural algorithm is used to select N top views among all views. Experiments show that this proposed algorithm has lower cost and higher speed than genetic algorithm, cuckoo search algorithm and Differential algorithm.

Keywords: Data warehouse, materialized view, hybrid cultural algorithm.

تاریخ ارسال مقاله: ۱۳۹۴/۰۳/۲۲
تاریخ اصلاح مقاله: ۱۳۹۴/۰۴/۳۱
تاریخ پذیرش مقاله: ۱۳۹۴/۱۰/۱۴
نام نویسنده مسئول: نگین دانشپور
نشانی نویسنده مسئول: ایران - تهران - لویزان - خیابان شعبانلو - دانشگاه تربیت دبیر شهید رجایی - دانشکده مهندسی کامپیوتر.

۱- مقدمه

تکنولوژی پایگاه داده تحلیلی[۱] برای یکپارچه کردن داده‌ها از چندین پایگاه داده بزرگ و نامنسجم که در مکان‌های مختلفی قرار دارند به کار می‌رود. از این تکنولوژی در پیاده‌سازی سیستم‌های تصمیم‌گیرنده و تحلیل‌گر استفاده می‌شود. ازآنجایی‌که سازمان‌های بزرگ دارای پایگاه‌های داده‌ای با حجم بسیار زیاد و پراکنده هستند، مدیران ارشد این سازمان‌ها برای کاربردهایی نظیر تصمیم‌گیری، سیاست‌گذاری و غیره نیاز به گرفتن گزارش از این پایگاه‌های داده و تحلیل آن‌ها دارند. داده‌های موردنیاز این پایگاه‌های داده، استخراج شده و به‌طور یکپارچه در مخزنی بنام پایگاه داده تحلیلی ذخیره می‌شوند[۱]. به سیستم‌هایی که از پایگاه داده تحلیلی استفاده می‌کنند سیستم‌های پردازش تحلیلی برخط[۲] گفته می‌شود. پرس‌وجوهایی که به سیستم‌های OLAP می‌رسد، بسیار پیچیده‌تر از پرس‌وجوهایی است که به سیستم‌های OLTP می‌رسد. یکی از مشکلات استفاده از پایگاه داده تحلیلی، سرعت پایین سیستم‌های OLAP در پاسخ‌گویی به پرس‌وجوها است. یکی از راه‌های افزایش سرعت پاسخ‌گویی به پرس‌وجوها در پایگاه داده تحلیلی ذخیره دید[۳]ها است؛ اما به دلیل کمبود حافظه و صرف زمان زیاد جهت به‌روزسازی این دیدها، امکان ذخیره‌سازی همه دیدها وجود ندارد. به همین علت باید تعدادی از دیدها برای پاسخ‌گویی مناسب به پرس‌وجوها انتخاب شده و ذخیره شوند [۲]. انتخاب یک مجموعه دید از بین کلیه دیدهای موجود معادل جستجو در یک فضای بسیار بزرگ است که یک مسئله NP hard به‌حساب می‌آید [۳] که دارای پیچیدگی معادل $O(2^n)$ است که n تعداد دیدها را نشان می‌دهد؛ بنابراین حل مسئله انتخاب دید با الگوریتم‌های ساده جستجو غیرممکن است. ازاین‌رو هم‌زمان با همه‌گیر شدن پایگاه‌های داده تحلیلی، الگوریتم‌هایی جهت انتخاب دیدهای مناسب ارائه شدند [۴].

اگرچه تاکنون هیچ الگوریتم قطعی برای حل مسئله انتخاب دید یافته نشده است، اما روش‌های زیادی برای بهبود حل این مسئله مطرح شده‌اند. مهم‌ترین پارامتر در انتخاب هر الگوریتم، پایین بودن هزینه پاسخ‌گویی به پرس‌وجوها با دیدهای انتخاب‌شده و درنتیجه سرعت بیش‌تر است. در این مقاله نیز روشی تکاملی بنام الگوریتم فرهنگی[۴] (CA) [۷]. برای حل این مسئله استفاده شده است. جهت تضمین مناسب بودن این الگوریتم، این الگوریتم با الگوریتم‌های ژنتیک[۵] و جستجوی فاخته[۶] مقایسه می‌شود.

مقاله‌ای که در پیش رو دارید به این شکل تقسیم‌بندی شده است: بخش ۲ مروری است بر کارهایی که قبلاً انجام شده است. در بخش ۳ این مقاله، به‌طور مختصر الگوریتم فرهنگی معرفی می‌شود و چگونگی پیاده‌سازی این الگوریتم روی مسئله انتخاب دید شرح داده خواهد شد. برای روشن شدن الگوریتم، در بخش ۴ مثالی از طریق انجام کار الگوریتم آورده شده است. در بخش ۵ آزمایش‌ها و مقایسه‌ها ارائه می‌شوند و در بخش ۶ نتیجه‌گیری انجام خواهد شد.

۲- پیشینه تحقیق

در یک تقسیم‌بندی کلی، می‌توان الگوریتم‌های انتخاب دید را در ۳ دسته کلی قرار داد: دسته اول که اولین الگوریتم‌های انتخاب دید بوده‌اند، ایستا بودند و دیدها قبل از اجرای پرس‌وجو انتخاب می‌شدند. روش حریصانه معروف‌ترین الگوریتم در این گروه است [۸]. روش کار این الگوریتم‌ها به این شیوه است که در هر تکرار دیدهایی با هزینه کم‌تر را انتخاب می‌کنند. ازآنجایی‌که جستجو در بین همه دیدها غیرممکن است، انتخاب این دیدها به‌صورت آفلاین انجام می‌شود؛ اما انتخاب دیدها به شکل آفلاین و بدون توجه به جریان کاری باعث می‌شود دیدها به شکل مناسبی انتخاب نشوند. این روش‌ها به دلیل ناکارآمد بودن، جای خود را به روش‌های پویا دادند [۶]

روش‌های پویا از سال ۲۰۰۰ برای اولین بار مطرح شدند. دو روش بعدی از روش‌های انتخاب دید، پویا هستند. روش‌های پویای انتخاب دید با توجه به پرس‌وجوی جاری سعی می‌کنند در کم‌ترین زمان بهترین مجموعه از دیدها را انتخاب کنند. دسته دوم الگوریتم‌های انتخاب دید، بر مبنای الگوریتم‌های پیش‌بینی‌کننده عمل می‌کنند. این الگوریتم‌ها با توجه به جریان کاری، پرس‌وجوهای آتی را پیش‌بینی کرده و برای آن‌ها دید مناسب انتخاب می‌کنند. ایراد اصلی این الگوریتم‌ها در مواردی است که جریان کاری قابل پیش‌بینی نیست. ازجمله این روش‌ها پیش‌بینی جریان کاری با استفاده از دسته‌بندی و پیش‌بینی جریان کاری بر اساس داده‌کاوی است [۹، ۱۰].

دسته سوم الگوریتم‌های انتخاب دید، بر مبنای الگوریتم‌های تکاملی[۷] هستند که عموماً برای حل مسائل پیچیده جستجو استفاده می‌شوند. این الگوریتم‌ها قادر به حل مسائل NP هستند. الگوریتم‌های تکاملی از یک مجموعه جواب تصادفی شروع کرده و با توجه به تابعی بنام تابع هزینه، راه‌حل را به جواب نسبتاً بهینه‌ای همگرا می‌کنند. جوابی بهینه خواهد بود که هزینه کم‌تری داشته باشد. (تابع هزینه در بخش ۲-۲-۳ به تشریح توضیح داده شده است.) الگوریتم‌های تکاملی به دلیل سرعت زیاد و عدم وابستگی به جریان کاری، جایگاه ویژه‌ای دارند، [۱۳-۱۱] . بااین‌وجود الگوریتم‌های تکاملی مشکلاتی نیز دارند. اصلی‌ترین ایراد این الگوریتم‌ها این است که به دلیل تصادفی بودن، تعدادی راه‌حل بی‌استفاده نیز تولید می‌شود که هزینه نگه‌داری را بالا می‌برد. برای حل این مشکل الگوریتم‌هایی نیز مطرح شده‌اند که از ترکیب الگوریتم تکاملی و یک الگوریتم دیگر به وجود آمده‌اند.

الگوریتم ژنتیک اولین الگوریتم تکاملی استفاده‌شده برای انتخاب دید بود که در سال ۱۹۹۹ توسط ژانک و یانک مطرح شد. پس‌ازآن، الگوریتم ژنتیک با تغییراتی مجدداً برای حل مسئله ژنتیک استفاده شد. بیش‌تر این تغییرات بر روی فضای اولیه مسئله اعمال شد تا سرعت همگرایی الگوریتم ژنتیک بالاتر برود و جواب‌های ناکارآمد کم‌تری تولید شود [۱۷-۱۴]. به دلیل نتیجه مطلوب الگوریتم ژنتیک، الگوریتم‌های تکاملی دیگری نیز مطرح شدند.

الگوریتم تپه‌نوردی[۸] که در سال ۲۰۱۳ مطرح شد، تعدادی جواب تصادفی به‌عنوان مجموعه راه‌حل اولیه انتخاب می‌کند و بر اساس تابع هزینه از میان بهترین جواب‌ها و همسایگان آن‌ها جواب بهتر را انتخاب می‌کند. سرعت همگرایی این الگوریتم وابستگی زیادی به مجموعه جواب تصادفی که در آغاز الگوریتم انتخاب می‌شود دارد [۱۸].

الگوریتم جهش قورباغه[۹] در سال ۲۰۱۰ برای حل مسئله انتخاب دید مطرح شده است. این الگوریتم جمعیت اولیه را به چند گروه مجزا تقسیم می‌کند. در مسئله انتخاب دید هر قورباغه یک راه‌حل برای مسئله است. بر اساس این روش اول جستجوی محلی انجام می‌شود و بر اساس آن قورباغه‌ها در هر گروه با تبادل اطلاعات، موقعیت خود را نسبت به غذا (بهترین جواب) بهبود می‌دهند و سپس تبادل اطلاعات بین گروه‌ها انجام می‌شود که بر اساس آن، بعد از هر جستجوی محلی در گروه‌ها، اطلاعات به‌دست‌آمده بین گروه‌ها باهم مقایسه می‌شود. سپس جستجوی محلی برای جهش قورباغه‌های با بدترین شایستگی به سمت قورباغه‌های با بهترین شایستگی صورت می‌پذیرد [۱۹]. این الگوریتم به دلیل محاسبات زیاد، دارای هزینه محاسباتی بالا است.

الگوریتم کلونی مورچگان[۱۰] نیز در سال ۲۰۱۰ مطرح شده است، در این الگوریتم هر مجموعه دید به‌عنوان یک راه‌حل و به یک مورچه اطلاق می‌شود. در هر تکرار الگوریتم مورچه‌ها به سمت مورچه‌ای که مسیر کم‌تر یعنی هزینه کم‌تری دارد همگرا می‌شوند [۲۰]. این الگوریتم نسبت به گرفتار شدن جواب در یک حلقه تکراری و پیدا کردن کمینه محلی به‌جای کمینه عمومی حساس است.

الگوریتم شبیه‌سازی حرارت[۱۱] در سال ۲۰۱۰ برای انتخاب دید استفاده شده است، از یک جواب اولیه شروع می‌کند و سپس در یک حلقه تکرار به جواب‌های همسایه حرکت می‌کند. اگر جواب همسایه بهتر از جواب فعلی باشد، الگوریتم آن را به‌عنوان جواب فعلی قرار می‌دهد در غیر این صورت، الگوریتم آن جواب را با احتمالی به‌عنوان جواب فعلی می‌پذیرد. این احتمال تابعی از اختلاف هزینه بهترین جواب و جواب فعلی و دما است. سپس دما به‌آرامی کاهش داده می‌شود. در گام‌های اولیه دما خیلی بالا قرار داده می‌شود تا احتمال بیش‌تری برای پذیرش جواب‌های بدتر وجود داشته باشد. با کاهش تدریجی دما، در گام‌های پایانی احتمال کم‌تری برای پذیرش جواب‌های بدتر وجود خواهد داشت و بنابراین الگوریتم به سمت یک جواب خوب همگرا می‌شود [۲۱]. در این الگوریتم امکان کند شدن همگرایی در اثر کاهش تدریجی دما یا برعکس از دست دادن جواب بهینه به دلیل کاهش زیاد دما وجود دارد.

در الگوریتم حرکات پرندگان[۱۲] که در سال ۲۰۰۶ برای مسئله موردنظر مطرح شد، هر مجموعه دید، یک پرنده تعریف می‌شود. پرندگان به سمت مسیری پرواز می‌کنند که کوتاه‌ترین مسیر را برای رسیدن به غذا می‌پیماید. در این الگوریتم هر پرنده در حافظه خود بهترین مکانی که داشته است را حفظ می‌کند و همچنین بهترین جواب در بین کلیه راه‌حل‌ها هم ذخیره می‌کند. این الگوریتم پس از چندین تکرار به جواب نسبتاً بهینه همگرا می‌شود [۲۲].

الگوریتم شکل‌دهی رودخانه[۱۳] در سال ۲۰۱۴ مطرح شد و در اصل تکامل‌یافته الگوریتم کلونی مورچگان است. در این روش هر قطره نشانگر یک راه‌حل است و هدف رسیدن به دریا در کم‌ترین زمان است. در این روش نیز قطرات به‌تدریج به سمت مسیری با بیش‌ترین شیب که همان کم‌ترین هزینه است همگرا می‌شوند [۲۳].

الگوریتم جستجوی فاخته در سال ۲۰۱۵ برای حل مسئله انتخاب دید مطرح شد. در این الگوریتم هر مجموعه از دیدها یک تخم فاخته به‌حساب می‌آید و هدف این است که پس از گذشت چند نسل تخم‌ها در آشیانه‌های بهتری قرار بگیرند [۲۴].

الگوریتم تفاضلی[۱۴] با یک مجموعه تصادفی از راه‌حل‌ها شروع می‌کند و در هر تکرار یک جواب تصادفی از میان جمعیت انتخاب می‌کند و با جواب قبلی مقایسه می‌کند. درصورتی‌که این جواب هزینه کم‌تری داشته باشد با جواب قبلی جایگزین می‌شود و این فرایند تا رسیدن به جواب نسبتاً بهینه ادامه می‌یابد [۲۵] .

کلیه روش‌های تکاملی مطرح‌شده سعی داشته‌اند تا مقدار هزینه کلی پاسخ‌گویی به پرس‌وجوها را کاهش دهند. ازآنجایی‌که الگوریتم‌های تکاملی در حال پیشرفت و جایگزینی با الگوریتم‌های جدید و سریع‌تر هستند، در مسئله انتخاب دید نیز استفاده از روش‌های تکاملی جدیدتر نتیجه بهتر یعنی هزینه کم‌تر خواهد داشت.

در این مقاله از روشی تکاملی بنام الگوریتم فرهنگی برای انتخاب دیدهای مناسب از بین کلیه دیدها استفاده می‌شود. الگوریتم فرهنگی الگوریتمی است که برای بهینه‌سازی به کار می‌رود و از الگوریتم‌های دیگری همچون ژنتیک عملکرد بهتری دارد. نتایج آزمایش‌ها نشان می‌دهد الگوریتم فرهنگی در مقایسه با الگوریتم ژنتیک [۱۷]، الگوریتم جستجوی فاخته [۲۴] و الگوریتم تفاضلی [۲۵]، هزینه کلی (TVEC) کم‌تری دارد.

۳- الگوریتم پیشنهادی

انتخاب یک مجموعه دید مناسب از بین کلیه دیدهای کاندید، آن‌هم در شرایطی که تعداد ابعاد زیاد شود تقریباً غیرممکن است [۶-۱]. راهکار حل این مشکل، ذخیره‌سازی دید است؛ اما ازآنجایی‌که ذخیره‌سازی کلیه دیدها و نگه‌داری آن‌ها غیرممکن است، باید تعدادی دید برای ذخیره‌سازی انتخاب شود. جستجو در بین تعداد دیدهای زیادی که از ابعاد مختلف به دست آمده‌اند به زمان زیادی نیاز دارد. از قوی‌ترین الگوریتم‌های جستجو که می‌تواند برای مسئله انتخاب دید نیز مفید است الگوریتم فرهنگی است. الگوریتم فرهنگی یک الگوریتم بهینه‌سازی است درنتیجه برای انتخاب یک مجموعه جواب بهینه برای حل مسئله انتخاب دید روش مؤثری است. این الگوریتم یکی از الگوریتم‌های تکاملی به شمار می‌آید و همانند الگوریتم‌های تکاملی دیگر، با یک راه‌حل تصادفی شروع می‌کند. این الگوریتم در طی

چندین بار تکرار و حرکت در فضای مسئله، خود را به بهترین جواب که همان مجموعه‌ای از دیدهای با کم‌ترین هزینه است، نزدیک می‌کند. در حل این مسئله، از یک مکعب چندبعدی از دیدها که در بخش ۳-۱ شرح داده شده است، به‌عنوان ورودی استفاده می‌شود. درنهایت k تا از دیدها به‌عنوان جواب نسبتاً بهینه انتخاب می‌شود. ازاین‌رو به این شیوه حل مسئله انتخاب دید top-k گفته می‌شود.

در این بخش ابتدا شبکه چندبعدی شرح داده می‌شود و سپس الگوریتم فرهنگی به‌اختصار شرح داده می‌شود و الگوریتم پیشنهادی بر پایه الگوریتم فرهنگی معرفی می‌شود.

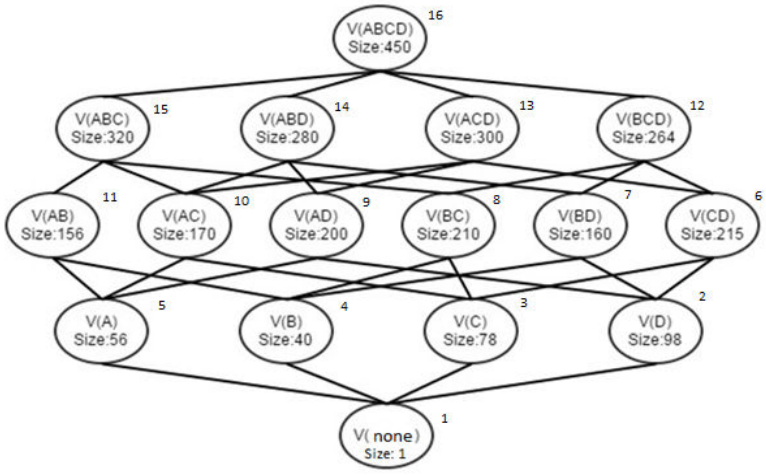
۳-۱- شبکه چندبعدی

یک شبکه چندبعدی مجموعه‌ای از تمام دیدهای کاندید است. هر نود از این شبکه نمایانگر یک دید است. در هر نود شماره دید، اندازه دید و این‌که این دید خود از مجموع چه جداول پایه‌ای به دست آمده است ذخیره می‌شود. در شکل ۱ نمونه‌ای چهاربعدی از این شبکه آمده است. هر یال بین دو نود نمایانگر وابستگی بین دیدها است. به‌طورکلی دید A به دید B وابسته است اگر پرس‌وجوهایی که با A پاسخ داده می‌شوند با B نیز قابل پاسخ‌گویی باشند. به‌عنوان‌مثال دید V(BC) به دید V(ABC) وابسته است [۲۶، ۲۷].

۳-۲- الگوریتم فرهنگی

الگوریتم فرهنگی شبیه‌سازی از شیوه زندگی انسان‌ها است. در جامعه انسان‌ها یک فضای جمعیت و یک فضا از باورهای این جمعیت وجود دارد. انسان‌ها از باورهایشان تأثیر می‌گیرند و بر باورهایشان تأثیر می‌گذارند. افراد موفق‌تر در جمعیت، الگوی باور به‌حساب می‌آیند و انسان‌های دیگر سعی می‌کنند با تبعیت از باورهای افراد موفق‌تر به رشد و تعالی برسند. در الگوریتم فرهنگی نیز دو فضا تعریف شده است؛ فضای جمعیت و فضای باور. فضای جمعیت که در الگوریتم فرهنگی تعریف می‌شود دقیقاً مشابه فضای جمعیت در الگوریتم ژنتیک است؛ یعنی یک فضای کاملاً تصادفی به‌عنوان فضای جمعیت اولیه در نظر گرفته می‌شود. فضای باور، وجه تمایز بین الگوریتم ژنتیک با الگوریتم فرهنگی است. هر دو فضا به‌صورت موازی باهم کار می‌کنند و بر روی‌هم تأثیر می‌گذارند. یک پروتکل ارتباطی این دو فضا را به هم مرتبط می‌کند. الگوریتم فرهنگی به‌صورت زیر عمل می‌کند: در هر نسل ابتدا هزینه اعضای موجود در فضای جمعیت ارزیابی می‌شود. بر اساس این ارزیابی، اعضایی که شایستگی تأثیرگذاری بر فضای باور را دارند انتخاب می‌شوند. از تجربیات این اعضا، برای ساختن و تغییر فضای باور استفاده می‌شود. درنتیجه فرهنگی در فضای باور به وجود می‌آید. در مرحله بعد فرهنگ ایجادشده در فضای باور، بر روی فضای جمعیت تأثیر می‌گذارد. پس از گذشت چند نسل، فضای جمعیت با استفاده از فرهنگ شکل‌گرفته به سمت جمعیت ایده‌ال، یعنی جمعیتی با هزینه پایین همگرا می‌شود [۷].

سه بخش اصلی الگوریتم فرهنگی یعنی فضای جمعیت، فضای باور و پروتکل ارتباطی در شکل ۲ نشان داده شده است.



شکل ۱: شبکه ۴ بعدی از دیدها

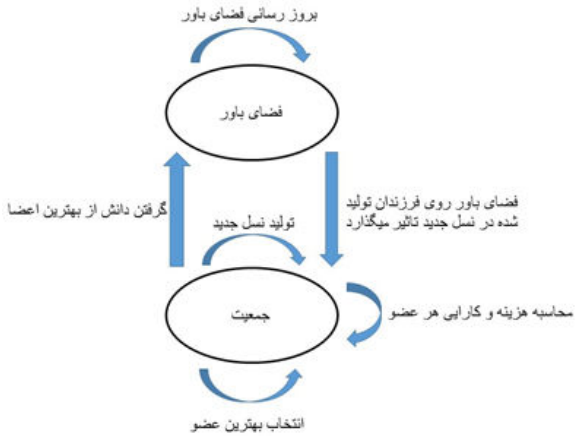
فضای باور نقش کلیدی در هدایت اعضای جمعیت در فضای جمعیت به سمت جواب بهینه ایفا می‌کند. فضای باور با ترکیب چندین منبع دانش که به ترتیب روی اعضا اثر می‌گذارند این کار را انجام می‌دهد. این منابع دانش شامل دانش موقعیتی[۱۵]، دانش هنجاری[۱۶]، دانش دامنه‌ای[۱۷]، دانش فضایی[۱۸] و دانش تاریخی یا موقت[۱۹] هستند که توسط رینالد پیشنهاد شده‌اند [۷]. به‌علاوه، هر دو فضا می‌توانند با یکدیگر توسط پروتکل ارتباط برقرار کنند. منظور از پروتکل درواقع دو تابع به نام‌های تابع صلاحیت و تابع تأثیر است. تابع صلاحیت، تابعی است که ورودی آن اعضای فضای جمعیت و خروجی آن، تغییر دادن مقادیر ذخیره‌شده توسط دانش‌های مختلف در فضای باور است. تابع تأثیر، تابعی است که ورودی آن از فضای باور و خروجی تغییر اعضای

فضای جمعیت، به‌منظور همگرایی به جواب نسبتاً بهینه است. جزئیات الگوریتم CA در شکل ۲ نشان داده شده است.

در ادامه به تشریح هر یک از این دو فضا و پروتکل ارتباطی پرداخته شده است.

فضای جمعیت

این فضا درواقع فضای اصلی جمعیت است. الگوریتم فرهنگی کار خود را با تشکیل یک جمعیت اولیه آغاز می‌کند. برای تشکیل جمعیت اولیه از یک تابع کاملاً تصادفی استفاده می‌شود.



شکل ۲: چارچوب الگوریتم فرهنگی [۷]

فضای باور

فضای باور، از تجربیات اعضای بهتر که از فضای جمعیت استخراج شده‌اند، شکل می‌گیرد. این تجارب در کلیه نسل‌ها شکل گرفته و ذخیره می‌شود. مبنای کار الگوریتم فرهنگی بر استفاده از این تجارب برای شکل‌دهی باور است. این باور در نسل‌های بعدی بر روی فضای جمعیت تأثیر می‌گذارد. ازآنجایی‌که اعضا موفق‌تر و بهتر از فضای جمعیت شانس بیش‌تری برای تأثیرگذاری بر روی فضای باور را دارند، به‌تدریج نسل‌های بعدی به سمت بهترین جواب همگرا می‌شوند. درواقع، این فضا در هرس کردن فضای جمعیت مؤثر است. هر عضو، یک ذره در فضای جستجو است و فضای باور برای دور ساختن اعضا از ناحیه‌های نامطلوب و سوق دادن آن‌ها به سمت ناحیه‌های امیدبخش و نزدیک به جواب به کار برده می‌شود.

دانش‌های مختلفی فضای باور را تشکیل می‌دهند. در کارهای پیشین انجام‌شده بر روی الگوریتم فرهنگی، تعدادی دانش ابداع و به فضای باور اضافه شده‌اند. در این مقاله از ۴ دانش برای تشکیل فضای باور موردنظر استفاده شده است. در ادامه این ۴ دانش توضیح داده شده است.

- دانش هنجاری: مجموعه‌ای از مقادیری است که ازنظر فضای مسئله، ارزشی مطلوب برای اعضای جمعیت به‌حساب می‌آید. به‌طور مثال در یک مسئله بهینه‌سازی، ارزش مطلوب هزینه کم‌تر برای اعضای فضای مسئله است.
- دانش موقعیتی: هر رویداد یا مقادیری از بین اعضای جمعیت که برای حل مسئله بااهمیت به‌حساب بیاید توسط این دانش ذخیره می‌شود. به‌عنوان‌مثال این دانش مقدار کمینه یا بیشینه اعضای یک جمعیت را در مسائل بهینه‌سازی ذخیره می‌کند.
- دانش تاریخی: ممکن است لازم باشد اطلاعاتی از فضای جمعیت ذخیره شود. این اطلاعات توسط دانش تاریخی ذخیره می‌شود تا در نسل‌های بعدی اجرای الگوریتم مورداستفاده قرار گیرد.
- دانش فضایی: اطلاعـاتی شامل توپوگرافی فضای جمعیت ارائه می‌دهد. این اطلاعات می‌تواند شامل تقسیم‌بندی فضای مسئله برای همگرایی سریع‌تر به جواب باشد.

از این چهار دانش برای تأثیرپذیری و همچنین تأثیرگذاری فضای باور از فضای جمعیت استفاده می‌شود. هر یک از این دانش‌ها بسته به نوع مسئله اطلاعات خاصی را ذخیره می‌کند. چگونگی مقدارگیری این دانش‌ها در الگوریتم پیشنهادی در بخش ۳-۳-۶ آمده است.

۳-۲-۱- پروتکل ارتباطی

الگوریتم فرهنگی نیازمند یک واسط میان جمعیت و فضای باور است. اعضای بهتر از فضای جمعیت، می‌توانند با استفاده از تابع به‌روزرسانی، فضای باور را به‌روز کنند. همچنین، دسته‌های دانش فضای باور می‌توانند توسط تابع اثرگذاری بر مؤلفه جمعیت اثر بگذارند. تابع اثرگذاری می‌تواند با تغییر فعالیت‌های اعضا، بر فضای جمعیت اثرگذار باشد.

۳-۲-۲- شبه کد الگوریتم فرهنگی

به‌طورکلی الگوریتم فرهنگی شامل مراحلی است که در ادامه توضیح داده شده است.

- فضای جمعیت را مقداردهی اولیه کن (جمعیت اولیه به‌طور تصادفی انتخاب می‌شود)
- فضای باور را مقداردهی اولیه کن (مقادیر اولیه برای دانش‌های مختلف را تعیین کن).
- موارد زیر را تکرار کن تا به شرایط پایان دادن برسی:
- فعالیت اعضا در فضای جمعیت را ارزیابی کن.
- هر عضو را با استفاده از تابع صلاحیت ارزیابی کن.
- والدین را جهت تولید نسل جدید انتخاب کن.
- به فضای باور اجازه بده تا ژنوم نسل جدید را با استفاده از تابع اثرگذاری تغییر دهد.
- فضای باور را با استفاده از تابع پذیرش به‌روز کن.
- اگر تعداد نسل‌ها به تعداد از پیش تعیین‌شده توسط ورودی الگوریتم بود و یا اینکه بهترین جواب تغییر نکرد، الگوریتم پایان می‌پذیرد.

بر این اساس، شبه‌کد الگوریتم عمومی فرهنگی در شکل ۳ آمده است [۷، ۲۸].

```
Procedure CA
Begin
        Time t← 0;
        Initialize the population P(t);
        Initialize the belief space B(t);
        Repeat
                Evaluate P(t);
                Accept best individuals;
                Update B(t) to build new belief space B'(t);
                Influence P(t) from p(t-1) and p'(t-1);
Until termination condition is met;
END
```

شکل ۱: شبه‌کد الگوریتم فرهنگی عمومی

۳-۳- معرفی الگوریتم پیشنهادی

به‌منظور استفاده از الگوریتم فرهنگی در مسئله انتخاب دید، الگوریتم جدیدی به نام الگوریتم ترکیبی (Hybrid CA) معرفی می‌شود. Hybrid CA (HCA) بر اساس الگوریتم فرهنگی عمل می‌کند؛ بنابراین قسمت اصلی از HCA دقیقاً مشابه الگوریتم فرهنگی است. با این تفاوت که یک تابع به آن‌ها اضافه می‌شود. یک تابع جدید به نام OPTIMIZATION() در داخل چارچوب قرار گرفته است [۷، ۲۸]. در انتهای بخش شبه‌کد الگوریتم HCA در شکل (۴) آورده شده است.

همان‌طور که در بخش قبل گفته شد برای تعریف الگوریتم پیشنهادی لازم است مواردی تعیین شوند که در زیر بخش‌های زیر به تعیین آن‌ها می‌پردازیم.

۳-۳-۱- نحوه تعریف و نمایش جمعیت اولیه:

ازآنجایی‌که روش استفاده‌شده برای حل مسئله انتخاب دید روش top-k است، برای تعریف هر عضو از جمعیت اولیه از یک زیرمجموعه k تایی از دیدها استفاده می‌شود. هرکدام از این اعضا راه‌حل‌هایی برای مسئله خواهند بود. برای نشان دادن هر مجموعه‌ای از اعضا که به‌عنوان راه‌حل انتخاب می‌شود، از یک نمایش مجموعه‌ای استفاده می‌شود. به این شکل که (۱،۴،۵،۱۱،۶،۱۰،۱۴) به معنی انتخاب دیدهای ۱ و ۴ و ۵ و ۱۱ و ۶ و ۱۰ و ۱۴ است.

۳-۳-۲- ارزیابی راه‌حل‌ها:

به‌منظور بررسی مناسب بودن هر راه‌حل به تابع هزینه‌ای نیاز است که هزینه پاسخ‌گویی به پرس‌وجوها را با توجه به دیدهای انتخاب‌شده محاسبه کند. هرچه این هزینه کم‌تر باشد راه‌حل مناسب‌تر است. تابع TVEC که در رابطه ۱ آمده است تابع هزینه است که با استفاده از اندازه دیدهای ذخیره‌شده پاسخگو به پرس‌وجوها محاسبه می‌شود. در رابطه TVEC، N تعداد کل دیدهای موجود در شبکه است و SM_v وضعیت هر دید است. درواقع اگر دید ذخیره‌شده باشد SM_v آن دید برابر با صفر است و اگر ذخیره نشده باشد برابر با یک خواهد بود. sizeSMA(v_i) اندازه کوچک‌ترین جد ذخیره‌شده دید i است.

فرض کنید برای پاسخ‌گویی به پرس‌وجویی مثل Q باید از دیدهای کاندید موجود در یک شبکه چندبعدی استفاده کرد. برای محاسبه هزینه کلی پاسخ‌گویی به Q از رابطه هزینه TVEC استفاده می‌شود. در محاسبه هزینه هر راه‌حل ابتدا هزینه محاسبه دیدهایی که قبلاً ذخیره شده‌اند محاسبه می‌شود. ازآنجایی‌که این دیدها از قبل ذخیره شده‌اند پس هزینه کلی پاسخ‌گویی به پرس‌وجوها توسط این دیدها فقط شامل اندازه هر دید است. همان‌طور که مشخص است، سطح پایین سیگمای اول برابر با $i = 1 \wedge SM_{V_i} = 1$ است؛ یعنی دیدهایی که دارای SM_{V_i} برابر با یک باشند؛ و نشان‌دهنده این است که این دید ذخیره شده است. به همین دلیل در سیگمای اول فقط اندازه‌ها باهم جمع بسته می‌شود. سیگمای دوم روی $i = 1 \wedge SM_{V_i} = 0$ جمع بسته می‌شود. در این رابطه SM_{V_i} برابر با صفر است، به این معنی که این دیدها هنوز ذخیره نشده‌اند. برای این دیدها، باید هزینه کوچک‌ترین جد هر دید که قبلاً ذخیره شده است، در نظر گرفته شود چراکه برای دسترسی به دیدی که ذخیره نشده است، باید از طریق اجداد آن اقدام کرد.

مزیت محاسبه هزینه به این روش این است که باعث می‌شود انتخاب دید برای هر پرس‌وجو به پرس‌وجوهای قبلی وابسته نباشد.

$$TVEC = \sum_{i=1 \wedge SM_{V_i}=1}^{N} size(V_i) + \sum_{i=1 \wedge SM_{V_i}=0}^{N} sizeSMA(V_i) \quad (۱)$$

۳-۳-۳- تعریف و مقداردهی به فضای باور

برای مقداردهی به فضای باور، باید با استفاده از پروتکل ارتباطی، دانش‌های مختلف در فضای باور مقداردهی شوند. مقداردهی به این دانش‌ها در مسئله موردنظر به شکل زیر است:

دانش موقعیتی: بهترین عضو از فضای جمعیت، انتخاب شده و مقدار آن در این دانش قرار می‌گیرد. در مسئله انتخاب دید، بهترین عضو، راه‌حلی با کم‌ترین هزینه است.

دانش هنجاری: برخلاف دانش موقعیتی، این دانش مقدار بیش‌ترین هزینه را از بین هزینه‌ها انتخاب و نگه‌داری می‌کند. از این دانش در نسل‌های بعدی برای مشخص کردن محدوده‌ای که توسط دانش فضایی تقسیم‌بندی می‌شود، استفاده می‌شود. هرچه نسل‌های بیش‌تری بگذرد مقدار این بیشینه کم‌تر و درنتیجه فضای جستجو کوچک‌تر خواهد شد.

دانش فضایی: محیط جستجو را به ناحیه‌هایی تقسیم می‌کند که مرزهای آن توسط دانش هنجاری و دانش موقعیتی تعیین می‌شود. (در اینجا ما از ۱۰ ناحیه استفاده کرده‌ایم)؛ یعنی بر اساس بیش‌ترین و کم‌ترین مقدار، فضای مسئله را به ۱۰ قسمت تقسیم می‌کند. اطلاعات مربوط به این تقسیم‌بندی توسط دانش فضایی ذخیره می‌شود. هر ناحیه یک لیست از اعضا دارد که به آن ناحیه تعلق دارند و حالت هر ناحیه با میانگین هزینه‌های اعضای آن ناحیه مشخص می‌شود. با افزایش تعداد تکرارها در الگوریتم، فضای باور کوچک‌تر می‌شود و درنتیجه دیدها به مجموعه دید مناسب همگرا می‌شوند.

دانش تاریخی: جدیدترین تغییر در دانش هنجاری و زمان به‌روزرسانی دانش هنجاری را ذخیره می‌کند. نقش اصلی این دانش تشخیص مواقعی است که الگوریتم در یک کمینه محلی گیر افتاده است. با توجه به اطلاعات ذخیره‌شده توسط این دانش، این موقعیت قابل‌حل شدن است.

۳-۳-۴- تابع صلاحیت

تابع صلاحیت تصمیم می‌گیرد که کدام اعضا بتوانند روی فضای باور تأثیر بگذارند. در مسئله انتخاب دید، فقط بهترین راه‌حل توسط تابع صلاحیت انتخاب می‌شود و برای تابع تأثیر فرستاده می‌شود.

۳-۳-۵- تابع تأثیر

فضای باور با استفاده از تابع تأثیر بر روی اعضایی فضای جمعیت، تأثیر می‌گذارند. میزان این تأثیر بر هر عضو به کارایی آن راه‌حل وابسته است. به این معنی که اعضایی با کارایی بهتر، شانس بیش‌تری برای تأثیرپذیری از سمت فضای باور را دارند که باعث می‌شود سرعت همگرایی به سمت جواب بهینه سریع‌تر شود و راه‌حل‌های غیر کارآمد به‌سرعت حذف شوند. هر منبع دانش برای این‌که روی اعضا اثر بگذارد از متد چرخ رولت استفاده می‌کند تا میزان تأثیر بر اساس کارایی هرکدام از اعضا به دست بیاید. درنتیجه، اعضایی که دارای هزینه کم‌تری هستند تأثیرگذاری بیش‌تری خواهند داشت. برای این کار ابتدا باید میانگین اعضای در هر ناحیه مشخص شود. این میانگین از رابطه ۲ محاسبه می‌شود.

$$F_{Avg_i}=\frac{\sum_{j=1}^{k} f\left(x_j^{\,i}\right)}{k} \qquad (۲)$$

در رابطه ۲، x_j^i یعنی راه‌حل jام در iامین ناحیه، تعداد راه‌حل‌هایی است که در ناحیه iام وجود دارد و $f(x_j^i)$ مقدار هزینه راه‌حل j در ناحیه iام است.

بر اساس هزینه هر راه‌حل در روی چرخ رولت، به هر راه‌حل منطقه‌ای روی چرخ رولت اختصاص داده می‌شود. این محدوده با مقدار F_{Area} مشخص می‌شود. F_{Area} از رابطه ۳ قابل‌محاسبه است.

$$F_{Area_i}=\frac{F_{Avg_i}}{\sum_{j=1}^{n} F_{Avg_j}} \qquad (۳)$$

به دلیل این‌که در مسئله انتخاب دید، هزینه کم‌تر به معنای راه‌حل بهتر است، رابطه بالا به رابطه ۴ تغییر می‌کند. پس به‌جای استفاده از F_{Avg} از $\frac{1}{F_{Avg}}$ استفاده می‌شود.

$$F_{Area_i}=\frac{\frac{1}{F_{Avg_i}}}{\sum_{j=1}^{n}\frac{1}{F_{Avg_j}}} \qquad (۴)$$

در این رابطه F_{Area} درصدی از چرخ رولت که به ناحیه iام اختصاص پیدا می‌کند را مشخص می‌کند.

۳-۳-۶- تولید نسل جدید:

برای تولید نسل جدید از ۲ عملگر تقاطع یکنواخت[۲۰] و جهش استفاده می‌شود:

۳-۳-۷- عملگر تقاطع یکنواخت:

وقتی والدین فرزندی را تولید می‌کنند، اطلاعات از والدین به‌صورت زیر به فرزند می‌رسد. به‌عنوان‌مثال فرض کنید دو والد $P1$ و $P2$ با مقادیر زیر هستند:

P_1=(۱، ۴، ۵، ۱۱، ۶، ۱۰، ۱۴، ۱۳)

P_2=(۳، ۵، ۷، ۱۵، ۱۱، ۱۰، ۲، ۹)

در حل مسئله انتخاب دید با استفاده از الگوریتم فرهنگی، از روش پیوند تقاطع یکنواخت استفاده می‌شود. به‌این‌ترتیب که برای فرزند یک آرایه به طول والدین قرار داده می‌شود. یک آرایه موقت ماسک نیز به همان طول ساخته می‌شود. آرایه ماسک یک آرایه تصادفی از صفرها و یک‌ها است. در ابتدا آرایه فرزند خالی است؛ که به‌طور تصادفی پر می‌شود. همان‌طور که در جدول ۱ مشخص است، آرایه فرزند به این صورت پر می‌شود که در خانه‌ای از آرایه که ماسک صفر است، مقدار فرزند برابر $P1$ و در خانه‌ای که مقدار ماسک یک است مقدار فرزند برابر $P2$ خواهد بود. درنتیجه فرزند حاصل، برابر خواهد بود با: C= (۱، ۴، ۷، ۱۵، ۶، ۱۰، ۲، ۱۳).

جدول ۱: نحوه تولید فرزند از ۲ والد با استفاده از الگوریتم تقاطع یکنواخت

راه‌حل‌ها	لیست دیدهای انتخاب‌شده							
P1	۱	۴	۵	۱۱	۶	۱۰	۱۴	۱۳
P2	۳	۵	۷	۱۵	۱۱	۱۰	۲	۹
ماسک	۰	۰	۱	۱	۰	۱	۱	۰
فرزند	۱	۴	۷	۱۵	۶	۱۰	۲	۱۳

۳-۳-۸- عملگر جهش

برای این‌که الگوریتم پس از مدتی در بین یک سری رشته جواب گیر نیفتد بهتر است از یک ضریب جهش استفاده شود. اگر ضریب جهش را ۰/۲ انتخاب کنیم، همواره ۲۰٪ از جواب‌ها با روش زیر جهش پیدا می‌کنند:

اگر راه‌حلی به شکل P_1=(۱، ۴، ۵، ۱۱، ۶، ۱۰، ۱۴، ۱۳) وجود داشته باشد، خانه شماره چهارم آن با عددی کاملاً تصادفی جایگزین می‌شود: P_2=(۱، ۴، ۵، ۹، ۶، ۱۰، ۱۴، ۱۳).

۳-۳-۹- استراتژی انتخاب

فرض کنید که μ والد در جمعیت وجود دارد. تابع تأثیر در ابتدا فراخوانی می‌شود تا α فرزند تولید کند. سپس عملگرهای تقاطع و جهش اعمال می‌شوند تا λ فرزند تولید شود. درنتیجه به تعداد $(\mu+\alpha+\lambda)$ راه‌حل در جمعیت وجود دارد. وقتی تابع هزینه روی همه این اعضا اعمال شد به تعداد μ راه‌حل انتخاب می‌شود.

```
Procedure HCA
Begin
        Time t← 0;
        Initialize Population P(t);
Initialize Belief Space B(t);
        Optimize P(t) locally;
        Evaluate P(t);
        Repeat;
                Accept best individuals;
Update B(t) to build new belief space B'(t);
Influence P(t) to generate new population p'(t);
Recombine P'(t) to make new population P''(t);
T← t+1
Optimize P(t-1),P'(t-1) and P''(t-1) locally
Select P(t) from P(t-1),P'(t-1) and P''(t-1);
Until termination condition is met;
END
```

شکل ۴: شبه‌کد الگوریتم پیشنهادشده برای حل مسئله انتخاب دید با استفاده از الگوریتم فرهنگی

۳-۳-۱۰- تابع به‌روزرسانی

تابع به‌روزرسانی دانش فضای باور را به‌روز می‌کند. به‌روزرسانی فضای باور به معنی به‌روزرسانی فضاهای دانش است. در هر دور دانش موقعیتی برابر بهترین فرزند تولیدشده در آن نسل است. فضای هنجاری نیز بیش‌ترین مقدار را در خود ذخیره می‌کند. دانش فضایی مجدداً فضای مسئله را تقسیم می‌کند و فضای موقت مقدار بهترین قبلی را در خود ذخیره می‌کند.

۴- مثال

یک شبکه ۴بعدی مطابق آنچه در بخش ۳-۱ نشان داده شده است در نظر گرفته می‌شود. در این شبکه شماره هر دید در کنار نام هرکدام قرار گرفته است. اندازه هر دید در کنار هر نود شبکه نشان داده شده است. در این مثال، فرض شده است که قرار است ۹ دید از بین دیدها انتخاب شود. درواقع مسئله، یک مسئله *top - 9* است. در اولین قدم، یک جمعیت اولیه تعریف می‌شود. این جمعیت از ۱۰ راه‌حل تشکیل شده است . هر راه‌حل نشان‌دهنده مجموعه‌ای از دیدها است که این راه‌حل‌ها از شبکه ۴بعدی ورودی تولید می‌شوند. این جمعیت در جدول ۲ نشان داده شده است. برای نشان دادن هر دید از شماره آن دید که در شکل ۱ در کنار نود مربوط به هر دید نوشته شده است، استفاده شده است. به‌طور مثال ۱۲ نشان‌دهنده V(BCD) است. تابعی که در رابطه ۱ تعریف شده است، برای محاسبه هزینه هر راه‌حل استفاده می‌شود.

به‌عنوان نمونه TVEC راه‌حل ۱ یعنی(۱۱، ۱۵، ۵، ۹، ۱۴، ۱۰، ۳، ۱۶، ۱۲) در رابطه ۵ محاسبه شده است.

جدول ۲: جمعیت اولیه

راه‌حل‌ها	لیست دیدهای انتخاب‌شده								
راه‌حل ۱	۱۲	۱۶	۳	۱۰	۱۴	۹	۵	۱۵	۱۱
راه‌حل ۲	۷	۲	۱۲	۱۴	۲	۸	۶	۵	۱۱
راه‌حل ۳	۹	۷	۸	۱۵	۴	۹	۵	۲	۱۰
راه‌حل ۴	۶	۲	۳	۵	۴	۱۱	۱۲	۴	۹
راه‌حل ۵	۵	۶	۱۳	۱۵	۱۱	۲	۹	۳	۴
راه‌حل ۶	۷	۲	۹	۶	۱۴	۱۳	۳	۸	۵
راه‌حل ۷	۳	۶	۵	۱۲	۱۱	۹	۱۲	۷	۴
راه‌حل ۸	۱۴	۹	۷	۲	۶	۱۲	۱۱	۳	۸
راه‌حل ۹	۱۲	۳	۱۳	۱۰	۱۱	۶	۹	۵	۲
راه‌حل ۱۰	۱۰	۳	۱۲	۱۱	۷	۵	۱۴	۴	۲

$$\sum_{i=0}^{16} size(v_i) = size(12)+size(16)+size(3)+size(10)+size(14)+size(9)+size(5)+size(15)+size(11) =$$
$$264+450+78+170+280+200+56+320+156=1974$$
$$\sum_{k=0}^{16} sizeSMA(v_i) = sizeSMA(1)+sizeSMA(2)+sizeSMA(4)+sizeSMA(7)+sizeSMA(8)+sizeSMA(13)+sizeSMA(6) =$$
$$size(5)+size(9)+size(11)+size(12)+size(12)+size(16)+size(12) = \quad (۵)$$
$$56+200+156+264+264+450+264=1654$$
$$TVEC=1974+1654=3628$$

جدول ۳: هزینه محاسبه‌شده برای کلیه راه‌حل‌ها

راه‌حل‌ها	لیست دیدهای انتخاب‌شده									TVEC
راه‌حل ۱	۱۲	۱۶	۳	۱۰	۱۴	۹	۵	۱۵	۱۱	۳۶۲۸
راه‌حل ۲	۷	۲	۱۲	۱۴	۲	۸	۶	۵	۱۱	۳۰۱۲
راه‌حل ۳	۹	۷	۸	۱۵	۴	۹	۵	۲	۱۰	۲۴۵۹
راه‌حل ۴	۶	۲	۳	۵	۴	۱۱	۱۲	۴	۹	۳۴۲۵
راه‌حل ۵	۵	۶	۱۳	۱۵	۱۱	۲	۹	۳	۴	۳۳۰۶
راه‌حل ۶	۷	۲	۹	۶	۱۴	۱۳	۳	۸	۵	۳۷۴۴
راه‌حل ۷	۳	۶	۵	۱۲	۱۱	۹	۱۲	۷	۴	۳۹۴۳
راه‌حل ۸	۱۴	۹	۷	۲	۶	۱۲	۱۱	۳	۸	۲۱۹۰
راه‌حل ۹	۱۲	۳	۱۳	۱۰	۱۱	۶	۹	۵	۲	۲۲۳۵
راه‌حل ۱۰	۱۰	۳	۱۲	۱۱	۷	۵	۱۴	۴	۲	۳۵۴۸

همان‌طور که در رابطه ۱ گفته شده است، برای محاسبه هزینه باید تمام دیدهای کاندید را در نظر گرفت. این دیدهای کاندید در شبکه ۴بعدی ورودی آمده است و تعداد کلیه دیدها برابر با ۱۶ عدد است. همان‌طور که در رابطه ۵ مشخص است، در سیگمای اول، اندازه خود دیدهای۱۱، ۱، ۵، ۹، ۱۴، ۱۰، ۳، ۱۶، ۱۲ باهم جمع بسته می‌شود چراکه این دیدها از قبل ذخیره شده‌اند و SM_{vi} آن‌ها برابر با یک است. در سیگمای دوم برای دیدهایی که ذخیره نشده‌اند باید هزینه کوچک‌ترین جد هرکدام که ذخیره شده است استفاده شود. برای مثال کوچک‌ترین جد یا همان *sizeSMA(2)* برابر با اندازه دید شماره ۹ است و *sizeSMA(4)* برابر با اندازه دید ذخیره‌شده ۱۱ است. برای سایر دیدها نیز به همین ترتیب محاسبات انجام می‌شود که نتیجه در جدول ۳ نشان داده شده است.

بهترین راه‌حل، راه‌حلی است که کم‌ترین هزینه را دارد. این راه‌حل با برچسب راه‌حل بهتر مشخص می‌شود.

جدول ۴: بهترین راه‌حل

راه‌حل	لیست دیدهای انتخاب‌شده								
بهترین راه‌حل	۱۴	۹	۷	۲	۶	۱۲	۱۱	۳	۸

تا این مرحله، فضای جمعیت به‌طور کامل تعریف شده است و TVEC برای کلیه راه‌حل‌ها محاسبه شده است. قدم بعدی مقداردهی به فضای باور است. همان‌طور که در بخش ۳-۳-۳ گفته شد، فضای باور از ۴ منبع دانش استفاده می‌کند. در این مرحله هرکدام از فضاهای دانش باید مقداردهی شوند.

دانش موقعیتی: بهترین راه‌حل که همان راه‌حل با کم‌ترین TVEC است را مطابق با جدول ۴ نگه می‌دارد .

دانش هنجاری: راه‌حل‌هایی با کم‌ترین و بیش‌ترین TVEC را ذخیره می‌کند. این دانش، مقدار کم‌ترین ۲۱۹۰ و مقدار بیش‌ترین ۳۹۴۳ را ذخیره می‌کند.

دانش فضایی: فضای مسئله را به ۵ ناحیه تقسیم می‌کند و راه‌حل‌ها را بر اساس TVEC که دارند به ناحیه‌های مناسب ارتباط می‌دهد. در جدول ۵، این ۵ ناحیه مشخص شده‌اند. در هر ستون که مربوط به هر ناحیه است، راه‌حل‌های مناسب با آن ناحیه قرار گرفته‌اند.

جدول ۵: نواحی درنظرگرفته‌شده توسط دانش فضایی

-۳۵۹۴ ۳۹۴۵	۳۵۹۴-۳۲۴۳	-۲۸۹۲ ۳۲۴۳	-۲۵۴۱ ۲۸۹۲	۲۵۴۱-۲۱۹۰
راه‌حل ۱: ۳۶۲۸ راه‌حل ۶: ۳۷۴۴ راه‌حل ۷: ۳۹۴۳	راه‌حل ۴: ۳۴۲۵ راه‌حل ۵: ۳۳۰۶ راه‌حل ۱۰: ۳۵۴۸	راه‌حل ۲ ۳۰۱۲		راه‌حل ۳:۲۴۵۹ راه‌حل ۸: ۲۱۹۰ راه‌حل ۹: ۲۲۳۵

دانش تاریخی: تاریخچه بهترین راه‌حل‌ها را نگه می‌دارد. حفظ کردن کمینه‌ها در هر مرحله مانع گرفتار شدن در کمینه محلی می‌شود.

همان‌طور که در بخش ۳-۳-۵ ذکر شد، برای محاسبه میزان و چگونگی تأثیر فضای باور بر فضای جمعیت، ابتدا با استفاده از رابطه ۴ مقدار F_{Area} برای هر ۵ ناحیه مختلف محاسبه می‌شود

برای محاسبه F_{Area} لازم است ابتدا F_{Avg} مطابق رابطه ۲ محاسبه گردد. برای نمونه این کار را روی دسته اول انجام داده‌ایم که این عملیات در رابطه ۶ آورده شده است.

$$F_{Avg_2}=\frac{\sum_{j=1}^{3} f\left(x_j^2\right)}{3}=\frac{2459+2190+2235}{3}=2295 \qquad (۶)$$

محاسبات انجام‌گرفته در رابطه ۶ را برای هر ۵ ناحیه انجام می‌دهیم. نتیجه در جدول ۶ مشخص است.

جدول ۶: $\mathbf{F_{Avg}}$ محاسبه‌شده در هر ناحیه

F_{Avg_1}	F_{Avg_2}	F_{Avg_3}	F_{Avg_4}	F_{Avg_5}
۲۲۹۵	۰	۳۰۱۲	۳۴۲۶	۳۷۷۲

حالا با استفاده از F_{Avg} به‌دست‌آمده برای هر ناحیه، می‌توان F_{Area} را برای هر ناحیه محاسبه کرد. F_{Area} نشانگر درصدی است که روی چرخ رولت به هر ناحیه اختصاص پیدا می‌کند. به‌عنوان نمونه، این درصد را برای ناحیه ۲ حساب می‌کنیم. محاسبات در رابطه ۷ نشان داده شده است.

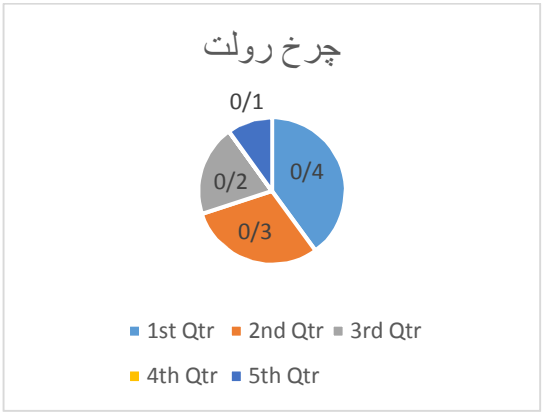
$$F_{Area_2}=\frac{\frac{1}{F_{Avg_1}}}{\sum_{j=1}^{n}\frac{1}{F_{Avg_j}}}=\frac{\frac{1}{2295}}{\frac{1}{2295}+\frac{1}{3012}+\frac{1}{3426}+\frac{1}{3772}}=0.4 \quad (۷)$$

مقدار محاسبه‌شده برای هر ناحیه در جدول ۷ ارائه شده است.

جدول ۷: درصد تخصیص‌داده‌شده روی چرخ رولت به هر ناحیه

F_{Area_1}	F_{Area_2}	F_{Area_3}	F_{Area_4}	F_{Area_5}
۰/۴	۰	۰/۲	۰/۳	۰/۱

شکل ۵ چرخ رولت استفاده‌شده را نشان می‌دهد. با استفاده از روش انتخاب چرخ رولت، ۲ ناحیه از بین ۵ ناحیه فوق انتخاب می‌شوند تا تابع تأثیر روی آن‌ها اثر بگذارد.



شکل ۵: چرخ رولت

پس از انتخاب دو ناحیه مطلوب برای اثرپذیری، بین همه اعضای این ۲ ناحیه و بهترین راه‌حل انتخاب‌شده، عملیات پیوند تقاطع یکنواخت و سپس جهش انجام می‌شود. با این کار، راه‌حل جدید به دست می‌آید و به راه‌حل‌های قبلی اضافه می‌شود.

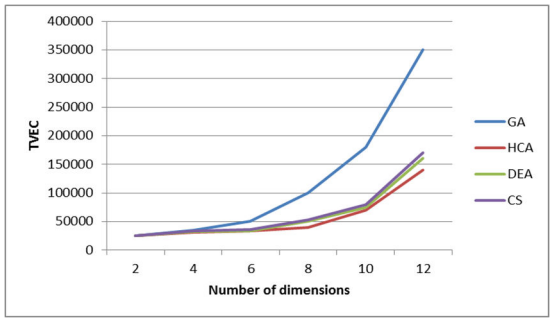
به‌عنوان‌مثال عملیات پیوند تقاطع یکنواخت که بین یکی از اعضای ناحیه دوم و بهترین راه‌حل انجام شده است در جدول ۸ نشان داده شده است.

جدول ۸: نمونه عملیات پیوند

راه‌حل‌ها	لیست دیدهای انتخاب‌شده								
بهترین راه‌حل (والد ۱)	۱۴	۹	۷	۲	۶	۱۲	۱۱	۳	۸
راه‌حل ۵ (والد ۲)	۵	۶	۱۳	۱۵	۱۱	۲	۹	۳	۴
ماسک	۱	۱	۰	۰	۰	۱	۰	۱	۱
راه‌حل جدید (فرزند)	۵	۶	۷	۲	۶	۲	۱۱	۳	۴

۵- آزمایش‌ها و نتایج

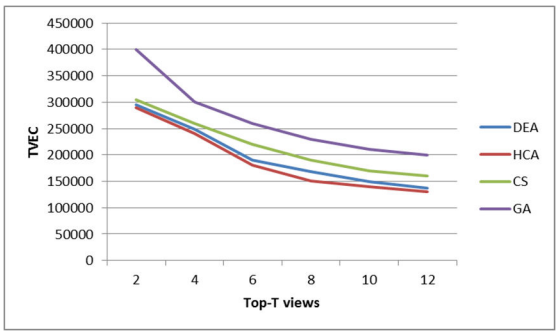
الگوریتم ارائه‌شده در سیستمی با مشخصات ذیل پیاده شده است. ویندوز ۷، ۲ گیگابایت حافظه رم، پردازنده ۲/۷۲ گیگاهرتز و چارچوب ویژوال استادیو. آزمایش‌ها با افزایش تعداد ابعاد انجام شده‌اند. در آزمایش‌های انجام‌شده، سه الگوریتم ژنتیک [۱۵]، جستجوی فاخته [۲۲]، الگوریتم تفاضلی [۲۳] و الگوریتم فرهنگی با یکدیگر مقایسه شده‌اند. در اولین نمودار که در شکل (۶) نشان داده شده است TVEC برای ۴ الگوریتم مذکور، در شرایطی که ابعاد افزایش پیدا کند، مقایسه شده است. ازآنجایی‌که افزایش تعداد ابعاد هزینه را به‌شدت زیاد می‌کند، تعداد ابعاد ۸ و بیش‌تر جزو مواردی است که شبکه چندبعدی دیدها را به یک ابرشبکه چندبعدی تبدیل می‌کند، درنتیجه افزایش تعداد ابعاد تا ۱۲ بررسی شده است. انتظار می‌رود برای ابعاد بیش‌تر نیز با توجه به سیر اکیداً صعودی نمودارها همین روند حفظ شود.



شکل ۶: مقایسه الگوریتم‌های **GA** و **CS** و **HCA**، درحالی‌که ابعاد افزایش می‌یابد

همان‌طور که در نمودارهای شکل ۶ نشان داده شده است، برای ابعاد بیش از ۴ بعد، TVEC مربوط به نمودارهای GA، CS و DEA در بالای نمودار HCA قرار می‌گیرند. درنتیجه TVEC انتخاب دید با استفاده از این الگوریتم در ابعاد بالا بسیار کم‌تر است.

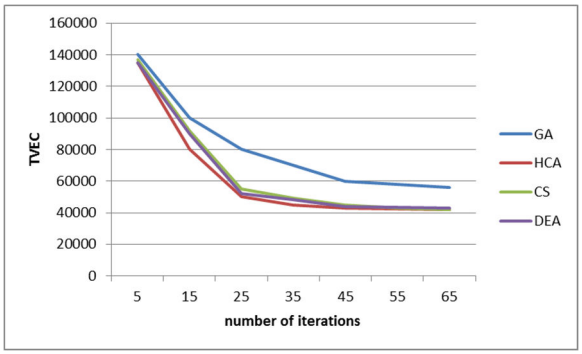
در نمودارهای رسم‌شده در شکل (۷)، تعداد دیدهایی که به‌عنوان راه‌حل انتخاب می‌شوند تغییر می‌کند. نتایج نشان می‌دهد افزایش تعداد دید انتخاب‌شده در هر چهار الگوریتم فوق باعث کم شدن هزینه‌ها، TVEC، می‌شود اما باوجود تغییر در تعداد دید انتخاب‌شده، همواره نمودار HCA زیر سه نمودار دیگر قرار دارد. به این معنی که تفاوت در تعداد دید انتخاب‌شده تأثیری در جواب آزمایش مبتنی بر برتری الگوریتم HCA نداشته است. همان‌طور که مشخص است، در الگوریتم HCA حتی اگر تعداد دیدهای ذخیره‌شده از ۱۰ بیش‌تر شود، دیگر تغییری در TVEC ایجاد نمی‌کند. پس در این آزمایش تعداد دیدها ۱۰ انتخاب می‌شود.



شکل ۷: مقایسه الگوریتم‌های **GA**، **CS** و **HCA**، درحالی‌که تعداد دیدهای ذخیره‌شده افزایش می‌یابد

در نمودارهای شکل ۸، تعداد تکرارهای الگوریتم متغیر است درحالی‌که تعداد دیدهای ذخیره‌شده و تعداد ابعاد ثابت است (تعداد

ابعاد ۱۰ و تعداد دیدها ۹ در نظر گرفته شده است). در این آزمایش با افزایش تعداد تکرار، در هر سه الگوریتم هزینه کاهش می‌یابد تا جایی که الگوریتم به جواب بهینه همگرا شود و تغییر دیگری نکند. این نمودار نشان می‌دهد که اولاً، در الگوریتم فرهنگی همگرایی سریع‌تر است. دوم این‌که نمودار هزینه در الگوریتم فرهنگی همواره در پایین دو الگوریتم دیگر قرار دارد که نشانگر کم‌تر بودن هزینه در کلیه شرایط مذکور نسبت به ۳ الگوریتم مقایسه‌شده دیگر است.



شکل ۸: مقایسه الگوریتم‌های **GA**، **CS** و **HCA**، درحالی‌که تعداد تکرارها افزایش می‌یابد

۶- نتیجه‌گیری

پایگاه داده تحلیلی، منبع داده‌ای برگرفته از منابع داده سازمان‌ها است که برای سیستم‌های تصمیم‌گیرنده، گزارش‌گیر و غیره استفاده می‌شود. پایگاه داده تحلیلی به دلیل حجم بسیار زیاد، دارای سرعت پایینی در پاسخ‌گویی به پرس‌وجوها است. یکی از راه‌های افزایش سرعت در این پایگاه داده‌های تحلیلی انتخاب دید است. اما نگه‌داری همه دیدها به حافظه زیاد نیاز دارد و هزینه نگه‌داری تعداد زیادی دید نیز بسیار بالاست. ازاین‌رو انتخاب تعدادی دید از بین دیدها و ذخیره آن‌ها اجتناب‌ناپذیر است.

در این مقاله روشی مبتنی بر الگوریتم فرهنگی برای انتخاب دید مناسب از بین دیدهای موجود در شبکه چندبعدی پیشنهاد شده است. الگوریتم فرهنگی از یک جمعیت تصادفی شروع می‌کند و پس از گذشت چند نسل به یک جواب مناسب متمایل می‌شود. این الگوریتم تا جایی که شرایط پایانی ارضا شود، تکرار می‌شود. در پایان الگوریتم، بهترین جواب به‌عنوان خروجی الگوریتم در نظر گرفته می‌شود که زیرمجموعه نسبتاً بهینه‌ای از دیدهای کاندید است. آزمایش‌ها نشان می‌دهد سرعت پاسخ‌گویی به پرس‌وجوها در پایگاه داده تحلیلی وقتی انتخاب دید با الگوریتم فرهنگی صورت بگیرد از الگوریتم‌های ژنتیک، جستجوی فاخته و الگوریتم تفاضلی بیش‌تر خواهد بود.

منابع

[1] Watson, H.J. and P. Gray, *Decision support in the data warehouse*, Prentice Hall Professional Technical Reference, 1997.

[2] Gupta, Himanshu. "Selection of views to materialize in a data warehouse." *In Database Theory—ICDT'97*, Springer Berlin Heidelberg, vol. 1186, pp. 98-112, 1997.

[3] Kumar, TV Vijay, Mohammad Haider, and Santosh Kumar. "A view recommendation greedy algorithm for materialized views selection." *In Information Intelligence, Systems, Technology and Management*, Springer Berlin Heidelberg, vol. 141, pp. 61-70, 2011.

[4] Baralis, Elena, Stefano Paraboschi, and Ernest Teniente. "Materialized Views Selection in a Multidimensional Database." *In VLDB*, vol. 97, pp. 156-165. 1997.

[5] Chirkova, Rada, Alon Y. Halevy, and Dan Suciu. "A formal perspective on the view selection problem." *The VLDB Journal—The International Journal on Very Large Data Bases 11*, no. 3,pp: 216-237, 2002

[6] Gupta, H., Mumick, I. "Selection of Views to Materialize in a Data Warehouse." *IEEE Transactions Knowledge and Data Engineering*, vol: 17 no:1, pp: 24–43, 2005

[7] Jin, Xidong, and Robert G. Reynolds. "Using knowledge-based evolutionary computation to solve nonlinear constraint optimization problems: a cultural algorithm approach." *In Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on*, vol. 3. IEEE, 1999.

[8] Vijay Kumar, T.V., Haider, M.: *A Query Answering Greedy Algorithm for Selecting Materialized Views*. In: Pan, J.-S., Chen, S.-M., Nguyen, N.T. (eds.) ICCCI 2010, Part II. LNCS (LNAI), Springer, Heidelberg,vol. 6422 pp. 153–162, 2010

[9] Daneshpour, Negin, and Ahmad Abdollahzadeh Barfourosh. "Dynamic View Management System for Query Prediction to View Materialization." *International Journal of Data Warehousing and Mining*, vol. 7,pp: 67-96, 2011.

[10] Aouiche, Kamel, and Jérôme Darmont. "Data mining-based materialized view and index selection in data warehouses." *Journal of Intelligent Information Systems*, vol. 33, no: 1,pp: 65-93, 2009.

[11] Horng, J.T., Chang, Y.J., Liu, B.J.: "Applying evolutionary algorithms to materialized view selection in a data warehouse. " *Soft Computing*, vol. 7, pp: 574–581, 2003.

[۱۲]میثم غلامی و جمال مشتاق. "الگوریتم ابتکاری کارآمد برای حل مسئله بازیابی در شبکه‌های توزیع الکتریکی با در نظر گرفتن حذف‌بار." مجله مهندسی برق دانشگاه تبریز، جلد ۴۴، شماره ۳، ص ۲۱-۱۳.

[۱۳]پدرام شهریاری‌نسب، معین پرستگاری و مهدی معلم. "استفاده از الگوریتم زنبورهای عسل برای بهینه‌سازی سیستم‌های انتقال توان بدون تماس به روش القایی برای شارژ خودروهای الکتریکی." مجله مهندسی برق دانشگاه تبریز، جلد ۴۳، شماره ۲، ص ۲۰-۹.

[14] Zhang, Chuan, and Jian Yang, "Genetic algorithm for materialized view selection in data warehouse environments." *Data Warehousing and Knowledge Discovery*, Springer Berlin Heidelberg, 1999, pp. 116-125.

[15] Goldberg, D.E., *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley (1989)

[16] Goldberg, D.E., Deb, K., "A comparative analysis of selection schemes used in Genetic Algorithms." *In: Foundations of Genetic Algorithms, MK*, pp. 69-93, 1991

[17] Zhou, L., X. He, and K. Li, "An improved approach for materialized view selection based on genetic algorithm." *Journal of Computers*, vol.7 no.7, pp. 1591-1598, 2012

[18] Kumar, TV Vijay, and Santosh Kumar. "Materialized view selection using iterative improvement." *In Advances in*

Computing and Information Technology, Springer Berlin Heidelberg, vol. 172, pp. 205-213, 2013

[19] Li, X., *et al.*, "Shuffled frog leaping algorithm for materialized views selection.". *in Education Technology and Computer Science (ETCS), 2010 Second International Workshop on*, 2010.

[20] Song, Xiangqian, and Lin Gao. "An ant colony based algorithm for optimal selection of materialized view." *In Intelligent Computing and Integrated Systems (ICISS), 2010 International Conference on, IEEE*, pp. 534-536., 2010.

[21] Derakhshan, R., et al., "Simulated Annealing for Materialized View Selection in Data Warehousing Environment.". *in Databases and Applications*, 2006.

[22] Wang, Yanni, Dibin Zhou, Yao Zheng, Kangjian Wang, and Tingjun Yang, "Viewpoint selection using PSO algorithms for volume rendering." *In Computer and Computational Sciences, 2007. IMSCCS 2007. Second International Multi-Symposiums on*, IEEE, pp. 286-291, 2007.

[23] Shayegh, P., Daneshpour, N.Materalized, "View selection using River Formation Dynamics Algorithm." *1st International Conference on computer sciences communication and information Technology*, pp 54-68, 2014.

[24] Shayegh, P.,Daneshpour N, "Materalized View selection using Cuckoo Search Algorithm." *20th conference of Computer Society of Iran*, pp. 374-378, 2015

[25] Vijay Kumar, T. V., and Santosh Kumar. "Materialised view selection using differential evolution." *International Journal of Innovative Computing and Applications*, vol. 6. no. 2 pp. 102-113, 2014.

[26] Harinarayan, V., Rajaraman, A., Ullman, J.D.: "Implementing data cubes efficiently." *In: ACM SIGMOD International Conference on Management of Data*, pp. 205-227, 1996.

[27] Lin, W.Y., Kuo, I.C., "A genetic selection algorithm for OLAP data cubes. Knowledge and Information Systems." vol. 6, no.1, pp: 83-102, 2004.

[28] Z. Xue and Y. Guo, "Improved Cultural Algorithm based on Genetic Algorithm." *in Proceedings of the IEEE International Conference on Integration Technology*, pp. 117-122, 2007.

زیرنویس‌ها

[1] Data Warehouse
[2] OnLine Analytical Processing (OLAP)
[3] View materialization
[4] Cultural algorithm
[5] Genetic algorithm
[6] Cuckoo search
[7] Evolutionary
[8] Hill climbing
[9] Shuffled Frog leaping
[10] Ant colonies
[11] Simulated Annealing
[12] Partial swarm optimization
[13] River Formation Dynamics
[14] Differential algorithm
[15] Situational knowledge
[16] Normative knowledge
[17] Domain knowledge
[18] Spatial knowledge
[19] Historical knowledge
[20] crossover