توسعه الگوریتم غذایابی کندوی زنبور عسل برای حل مسئله مسیریابی خودرو

امیرمسعود رحیمی (مسئول مکاتبات)، استادیار، گروه عمران، دانشکده مهندسی، دانشگاه زنجان، زنجان، ایران

احسان رمضانی‌خوانساری، دانشجوی دکتری، دانشکده عمران و محیط زیست، دانشگاه امیرکبیر، تهران، ایران

E-mail: amrahimi@znu.ac.ir

دریافت: ۱۳۹۲/۰۱/۲۰ پذیرش: ۱۳۹۳/۰۱/۳۱

چکیده:

یکی از موارد مسئله مسیریابی خودرو، پیدا کردن مسیری است که خودرو از یک نقطه حرکت را شروع می‌کند، سپس نقاط معین با مختصات ثابت را ملاقات کرده و درنهایت به نقطه آغازین باز می‌گردد. این مسئله یکی از مسائل سخت و پرکاربرد در حمل و نقل و علوم دیگر است که مسئله فروشنده دوره گرد نیز نامیده می شود. پیچیدگی این مسئله باعث ناکارآمدی روشهای قطعی در حل آن شده است، از این رو در این تحقیق سعی شده با استفاده از اصلاحاتی در الگوریتم کلونی زنبور عسل مصنوعی، که یکی از آخرین الگوریتم های ابتکاری است، به حل آن پرداخته شود. نتایج حاصل حاکی از توانایی الگوریتم کلونی زنبور عسل پیشنهادی برای حل مسئله مسیریابی خودرو در قیاس با سایر روش ها است.

واژه‌های کلیدی : مهندسی حمل و نقل، مسئله مسیریابی خودرو، الگوریتم کلونی زنبور عسل.

۱. مقدمه

مسئله مسیریابی خودرو (VRP[1]) یک مثال نمونه از مسائل بهینه‌سازی ترکیبی بسیار سخت است که انواع مختلفی دارد. یک نوع از مسئله مسیریابی خودرو که هدف آن ایجاد مسیری بسته به شرط ملاقات همه نقاط یک بار و فقط یک بار است، و به صورت عمومی مسئله فروشنده دوره گرد (TSP[2]) نامیده می‌شود. این مسئله نمونه‌های بسیاری در حمل و نقل مانند شرکت‌های سفارش کالا دارد ، مشتریان در طی روز به صورت غیرحضوری و از وب سایت، کالای خود را خریداری کرده و آن را برای تحویل در محل سکونت‌خود سفارش می‌دهند، شرکت سفارش‌های یک روز را جمع‌آوری کرده و در پایان روز خودروی حاوی سفارش‌ها باید از محل دپوی اجناس حرکت کرده و به محل همه مشتریان رفته و اجناس را تحویل دهد و سپس به دپو بازگردد. مثال دیگر، تریلرهای حمل زباله از مخازن سطح شهر است که از محل دپوی زباله‌ها خارج شده، سپس به تمام محل‌های ذخیره زباله رفته و باز به محل اول و دپوی اصلی مراجعت می‌کنند. مسئله فروشنده دوره‌گرد کاربردهای فراوانی در علوم دیگر مانند مخابرات برای چینش دکل‌های مخابراتی (BTS[3]) دارد. سؤال مهمی که در همه این مثال‌ها مطرح است را، می‌توان چنین بیان کرد که « چگونه می‌توان یک حلقه یا مسیر بسته با کمترین طول که منجر به کمترین هزینه می‌شود را یافت؟». مسئله مسیریابی خودرو متعلق به خانواده مسائل NP-Complete[4] است که پیچیدگی محاسبات آن با تعداد نقاط و ابعاد مسئله به صورت نمایی افزایش می‌یابد. بنابراین امروزه تمایل بسیاری به روش‌های ابتکاری برای حل مسئله مسیریابی خودرو ایجاد شده است [Geng, et al., 2011].

هدف از انجام این تحقیق آن است که با ایجاد تغییراتی در الگوریتم ابتکاری کلونی زنبور عسل مصنوعی جواب‌های بهینه و زمان حل مسئله مسیریابی خودرو را بتوان بهبود بخشید.

۲. مرور ادبیات

در دهه‌های اخیر با رشد فنآوری و ساخت رایانه‌های با توان پردازش بیشتر و فراگیری روش‌های نرم‌افزاری، و از سوی دیگر با کشف رفتار و تعامل طبیعی بین ارگانیزمها و حشرات، سعی بر آن شده است با استفاده از فنآوری و با الهام از رفتارهای موجود در طبیعت به حل مسائل پیچیده پرداخته شود. مسئله مسیریابی خودرو به وسیله روشهای ابتکاری قطعی و احتمالی مختلف آزموده شده و مورد حل قرار گرفته که عبارتند از :روش شاخه و کرانه و برش صفحات [Padberg and Rinaldi, 1991]، [Araque G., et al. 1994] [Fischetti, et al., 1997] و [Ascheuer, et al., 2000]، زنجیره‌ای مارکوف، برنامه نویسی دینامیکی، انواع شبکه‌های عصبی، [Masutti and Castro, 2009] , [Hasegawa, et al., 2002]، (Jemai and Mel-louli, 2008)، (Li, Yi, and Zhu, 2009) و (Ma, Yang, Hou, Tan, and Liu, 2008)، جستجوی ممنوعه (Fiechter, 1994)، (Barbarosoglu and Ozgur, 1999) و (Malek, Guruswamy, PANDYA, and OWENS, 1989) انواع مختلف روشهای الگوریتم ژنتیک [Potvin, 1996]، (Chat-terjee, Carrera, and Lynch, 1996)، (Abu Zitar, Essam, and Shehabat, 2007) و (Moon, Kim, Choi, and Seo, 2002) الگوریتم تبرید تدریجی شبیه سازی شده (Chiang and Russell, 1996) و (Zhao,dna, Geng, Chen, Yang, Shi 2011) و الگوریتم کلونی مورچه (Bel and McMullen, 2004)، (Ling, Hai-Ying, and Shu, 2012)، (Manfrin, Mauro, Stutzle, and Dorigo, 2006) .

یکی از روش‌های اخیر که کاربردهای قابل قبولی در مسائل گسسته و پیوسته داشته، الگوریتم کلونی زنبور عسل مصنوعی است که توسط یونزاوا و کیکوچی ارائه شد [Yonezawa and Kikuchi, 1996]. لوچیچ[5] و تئودورویچ[6] مطالعه بر روی الگوریتم کلونی

زنبور عسل را برای بررسی شش مسئله مسیریابی خودرو منتشر نمودند [Lucic and Teodorovic, 2003]. آنها کاربردهای ممکن هوش جمعی زنبور را در حل مسائل پیچیده ترافیک و مهندسی حمل و نقل مورد بررسی و اکتشاف قرار دادند. بهگد[۷] و پورانیک[۸] از الگوریتم کلونی زنبور عسل مصنوعی به منظور حل مسئله مسیریابی خودرو استفاده کردند. در این مطالعه از الگوریتم نزدیک‌ترین همسایه (NNA)[۹] به همراه الگوریتم کلونی زنبور مصنوعی استفاده شد [Bhagade & Puranik, 2012]. جی[۱۰] و وو[۱۱] در تحقیقی به حل مسئله مسیریابی خودرو با شرایط ظرفیت معین با تابع زمان سفر غیر مستقل از طریق بهبود الگوریتم کلونی زنبور عسل مصنوعی پرداختند [Ji & Wu, 2011]. وونگ[۱۲]و چانگ در این مطالعه از تکنیک الگوریتم بهینه‌سازی کلونی زنبورعسل به همراه قانون انتقال به صورت ناحیه‌ای[۱۳] استفاده کرده بودند و به وسیله روش ابتکاری 2-opt در نزدیک‌ترین همسایگی با شعاع ثابت (FRNN 2-Opt) [۱۴] دورهای ساخته شده بهینه می‌گشتند. در الگوریتم ارائه شده، بعد از تکمیل یک دور کامل، استراتژی پیرایش بر پایه تکرار [۱۵] بررسی می‌کند که آیا انتقال نیازمند اصلاح از طریق الگوریتم 2-opt در نزدیک‌ترین همسایگی با شعاع ثابت است [Wong and Chong, 2009]. تحقیقات متنوعی با استفاده از کلونی زنبور عسل و ترکیب آنها با سایر الگوریتم‌ها برای حل مسئله مسیریابی خودرو صورت گرفته است که نشانگر توانایی الگوریتم کلونی زنبور عسل مصنوعی است.

۳. مبانی الگوریتم کلونی زنبور عسل مصنوعی

به طور کلی در الگوریتم کلونی زنبور عسل مصنوعی سه عامل حل مسئله وجود دارد : زنبورهای دیده‌بان یا زنبورهای دیده‌ور[۱۶]، زنبورهای پیشاهنگ یا زنبورهای کارگر [۱۷]و زنبورهای تماشاگر[۱۸] که به همین ترتیب به حل مسئله می‌پردازند. زنبور دیده‌بان فضای حل مسئله را به صورت کورکوانه جستجو می‌کند و آغاز کننده الگوریتم است. زنبورهای پیشاهنگ با توجه به جواب‌های زنبورهای تماشاگر فضای حل مسئله را به چند بخش تقسیم می‌کنند و سعی دارند که جواب حاصل از بخش خود را بهبود بخشند، زنبورهای تماشاگر که به هیچ بخشی تعلق خاطر ندارند، فقط تلاش می‌کنند به کمک زنبورهای پیشاهنگی که پاسخ‌های بهتر دارند، بروند؛ و روند بهبود جواب‌ها در آن بخش را بهبود بخشند، یعنی هیچ محدودیتی برای تعداد حضور زنبورهای تماشاگر در یک بخش وجود ندارد. این روند تا آنجا تکرار می‌شود که یک زنبور پیشاهنگ دیگر نتواند جواب‌های حاصل از بخش خود را حتی با کمک زنبورهای تماشاگر بهبود بخشد و این یعنی که اکسترمم آن بخش به دست آمده است، در این هنگام جواب به دست آمده در کندو ذخیره شده و زنبور پیشاهنگ بخش خود را رها کرده و به یک زنبور دیده‌بان تبدیل می‌شود. این چرخه تا آنجا تکرار می‌شود که اکسترمم‌های مسئله یکی بعد از دیگری به دست آید. در نهایت بین این اکسترمم‌ها یک اکسترمم‌گیری انجام می‌شود که جواب نهایی خواهد بود. یکی از قابلیت‌های مهم الگوریتم کلونی زنبور عسل سادگی در بیان ایده اصلی آن است که با تغییر در هر بخشی می‌توان آن را برای مسئله مورد نظر خود تطبیق داد.

۴. معادل سازی مسئله مسیریابی خودرو و الگوریتم کلونی زنبور عسل مصنوعی

مسئله مسیریابی خودرو که در گرافی شامل نقاط و کمان‌ها بیان می‌شود عبارت است از به دست آوردن کوتاه‌ترین مسیر بر روی نقاطی که از هر نقطه یک بار و فقط یک بار عبور کرده و در نهایت به نقطه آغازین بازگشته و یک دور کامل ایجاد گردد. به منظور معادل سازی الگوریتم کلونی زنبور عسل و مسئله مسیریابی خودرو، زنبورها در فضای مسئله مسیریابی خودرو

تعریف می‌شوند که در آن نقاط در مسئله مسیریابی خودرو همان گل‌های مورد جستجو برای برداشت شهد توسط زنبورها هستند، طول مسیر حرکت خودرو با کیفیت شهدی که زنبورها برداشت می‌کنند به صورت معکوس متناسب است، یعنی هرچه شهد بیشتری در الگوریتم کلونی زنبور عسل مصنوعی جمع‌آوری شود، متناسب با آن، طول کمتری در مسئله مسیریابی خودرو به دست خواهد آمد.

همواره در الگوریتم کلونی زنبور عسل هدف جمع‌آوری بیشترین شهد است. در جدول (۱) زیر به صورت خلاصه معادل سازی کلی مسئله مسیریابی خودرو و مفهوم الگوریتم کلونی زنبور عسل بیان شده است.

جدول ۱. معادل سازی مسئله مسیریابی خودرو و الگوریتم کلونی زنبور عسل

مسئله مسیریابی خودرو	کلونی زنبور عسل
خودرو	زنبورهای پیشاهنگ
نقاط	گل
طول مسیر	معکوس کیفیت شهد

۵. داده‌های مسئله مسیریابی خودرو

داده‌های مسئله برخی از مثال‌های حل شده که شامل مختصات نقاط و بالتبع فواصل بین آنهاست در ضمیمه آمده است. این داده‌ها از سایت co-/groups/de.heidelberg-uni.iwr.www//:http tsp/TSPLIB95/software/mopt استخراج شده‌اند که منبع معتبری در زمینه مسائل مسیریابی خودرو است و توسط سایر نویسندگان مقالات نیز استفاده شده است (گروه تحقیقاتی این مقاله به منظور مدل‌سازی به دنبال جمع‌آوری داده‌های واقعی از شرکت‌های حمل و نقل مواد غذایی و کارخانجات تولید کننده مواد غذایی در ایران بودند اما تلاش‌های انجام گرفته به ثمر نرسید و در نتیجه از داده‌های فوق همچون بسیاری از تحقیقات مشابه، استفاده شد).

۶. بیان الگوریتم کلونی زنبور عسل پیشنهادی

فرآیندهای الگوریتم کلونی زنبور عسل پیشنهادی در یک چرخه تولید مسیرعبارتند از :

۱) حرکت زنبور دیده‌ور به وسیله الگوریتم نزدیک‌ترین همسایگی گسسته برای کمک به ساخت مسیر اولیه

۲) حرکت زنبورهای پیشاهنگ و تماشاگر برای جستجوی مسیرهای بهتر

۳) تصمیم‌گیری در کندو، که هر کدام در ادامه به تفصیل بیان شده‌اند.

۶-۱ فرآیند زنبور دیده‌ور با الگوریتم نزدیک‌ترین همسایگی گسسته پیشنهادی

همان طور که گفته شد زنبور دیده‌ور جواب اولیه را به صورت کورکورانه به دست می‌آورد، اما در این مقاله برای نخستین مرتبه الگوریتم نزدیک‌ترین شعاع همسایگی گسسته ابداع شد تا به الگوریتم زنبور عسل مصنوعی پیشنهادی کمک و با جواب‌های اولیه بهتری حل مسئله را آغاز کند. همچنین برخلاف روش‌های قبلی، اغلب، زنبور دیده‌ور یک پاسخ کامل به دست می‌آورد و در ادامه، زنبورهای دیگر آن را بهبود می‌بخشند، در اینجا زنبور دیده‌ور قسمتی از یک پاسخ کامل را به دست می‌آورد، یعنی تعدادی لینک که مستعد بهبود هستند و نه یک دور کامل. در الگوریتم کلونی زنبور عسل پیشنهادی ابتدا یک زنبور عسل به صورت تصادفی از یک گل (نقطه) شروع به حرکت می‌کند و با استفاده از الگوریتم نزدیک‌ترین همسایگی، به ایجاد کمان‌های میان گل‌ها (شهرها) پرداخته می‌شود. این زنبور که زنبور دیده‌ور نامیده می‌شود کاملاً بر طبق الگوریتم نزدیک‌ترین همسایگی حرکت نمی‌کند و در شرایطی که نزدیک‌ترین نقطه قبلاً رؤیت شده باشد، مانند الگوریتم نزدیک‌ترین همسایگی به نقطه بعدی از نظر کوتاهی کمان حرکت نمی‌کند، بلکه به صورت تصادفی به نقطه رؤیت نشده دیگری می‌جهد و از آنجا دوباره الگوریتم

نزدیک‌ترین همسایگی را اجرا می‌کند و همواره، از تولید یک حلقه بسته جلوگیری می‌کند. این فرآیند، الگوریتم نزدیک‌ترین همسایگی گسسته نامیده شد. لازم به ذکر است در فرآیند زنبور دیده‌ور کندو در ابتدا در هیچ نقطه‌ای یا شهری (در دستگاه مختصات) قرار ندارد و خارج از آن فرض شد.

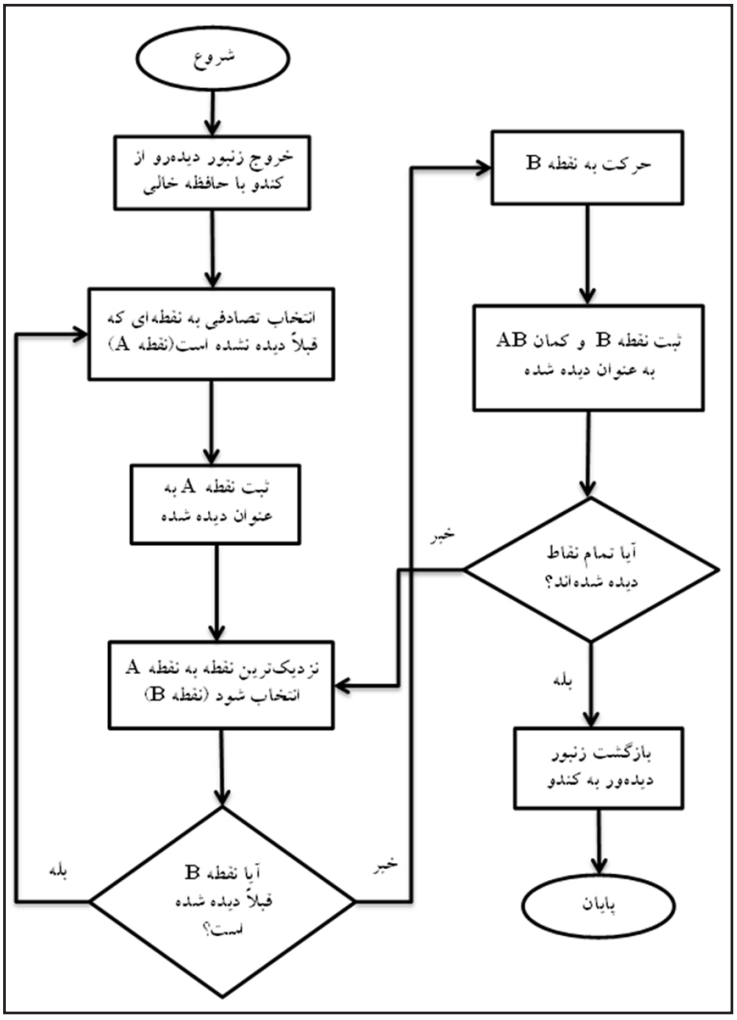
به طور کلی حرکت زنبور دیده‌ور در الگوریتم ابداعی نزدیک‌ترین همسایگی گسسته (این تحقیق) را در دو فاز می‌توان بیان کرد :

۱. حرکاتی که به صورت تصادفی انجام می‌گیرند و فقط هدف انتخاب نقطه بعدی به منظور انجام الگوریتم نزدیک‌ترین همسایه (NNA) است.

۲. حرکاتی که با استفاده از الگوریتم نزدیک‌ترین همسایه صورت می‌گیرد و هدف آن ایجاد کمان بین نقاط است. در شکل(۱)، فلوچارت ساخت قطعات پاسخ زنبور دیده ور با استفاده از الگوریتم نزدیک‌ترین همسایگی گسسته نشان داده شده است.

۶-۲ زنبورهای تماشاگر و پیشاهنگ مقید شده

بعد از فرآیند زنبور دیده‌ور، تمام اطلاعات کمان‌های ساخته شده توسط آنها به زنبورهای پیشاهنگ منتقل می‌شود و زنبورهای پیشاهنگ از کندو خارج می‌گردند که در این فرآیند، کندو به صورت تصادفی روی یک نقطه (گل) قرار دارد. زنبورهای پیشاهنگ وظیفه دارند بعد از بازدید تمام نقاط به کندو بازگشته و یک دور کامل ایجاد نمایند، یا به عبارتی یک پاسخ کامل به دست آورند. برخلاف الگوریتم‌های معمول زنبور عسل، در این



شکل ۱. فرآیند حرکت زنبور دیده ور

مقاله برای حرکت قید فاصله قرار داده شد و طبعاً روابط حاکم بر حرکت باتوجه به قید اصلاح شدند. حرکت میان نقاط در فرآیند زنبور عسل پیشاهنگ بر مبنای احتمال صورت می‌گیرد که تابع احتمال آن در روابط (۱)، (۲) و (۳) زیر آمده است.

$$P_i = \frac{\frac{1}{d_i}\left(\beta^{\left(\frac{d_{max}-d_i}{d_{max}-d_{min}}\right)^{\alpha}}\right)}{\sum\left[\frac{1}{d_i}\left(\beta^{\left(\frac{d_{max}-d_i}{d_{max}-d_{min}}\right)^{\alpha}}\right)\right]} \quad (۱)$$

$$i = \begin{Bmatrix} i\in A_c & ,A_c\neq 0 \\ i\in A_j & ,A_c = 0 \end{Bmatrix} \quad (۲)$$

$$C = Max\begin{Bmatrix} Scout_m \\ Best_m \end{Bmatrix} \quad (۳)$$

i: شماره (مشخصه) گل که احتمال حرکت به آن بررسی می‌شود

Pi: احتمال انتخاب گل i در حرکت بعدی

d: فاصله زنبور تا گل i

β: ضریب افزایش احتمال انتخاب (عدد دلخواه متناسب با مسئله)

d_{max} فاصله دورترین نقطه تا نقطه فعلی (با حذف نقاط رؤیت شده)

d_{min} فاصله نزدیک‌ترین نقطه تا نقطه فعلی (با حذف نقاط رؤیت شده)

α: مقدار ۰ یا ۱ نشانگر دیده شدن یا نشدن توسط زنبور دیده‌ور یا دور قبلی

j: نقطه‌ای که زنبور پیشاهنگ روی آن قرار دارد

A_j: مجموع نقاط دیده نشده توسط زنبور پیشاهنگ

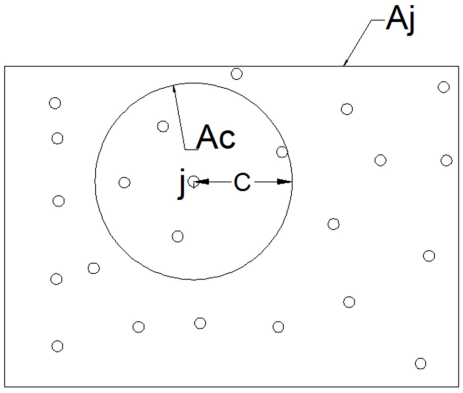
A_C: مجموع نقاط در شعاع C حول نقطه j

$Scout_m$: طول بزرگترین کمان به تولید شده توسط زنبور دیده‌ور در ابتدای چرخه

$Best_m$: طول بزرگ‌ترین کمان موجود در بهترین سراسری

C: شعاع مقید شده برای حرکت زنبور پیشاهنگ

این تابع احتمال هنگامی که زنبور پیشاهنگ در نقطه‌ای قرار دارد برای تمام نقاط اطراف آن در یک شعاع معین بررسی می‌شود. قیدی که برای حرکت زنبورهای پیشاهنگ قرار داده شده در شکل(۲) نشان داده شده است.



شکل ۲. محدوده حرکت زنبور پیشاهنگ مقید شده

همان طور که بیان شد α می‌تواند دو مقدار یک و صفر را اختیار کند. α =۱ نشان دهنده تأثیر رؤیت شدن کمان مورد بررسی توسط زنبور دیده‌ور (یا کمان دیده شده در بهترین دور قبلی که در دورهای دوم به بعد کاربرد دارد) است. حال ضریب β ، که ضریبی اختیاری و تجربی است استخراج می‌شود و از طرف کندو (توسط کاربر الگوریتم) به زنبور عسل پیشاهنگ القا و تحمیل می‌شود، این ضریب محدوده تاثیر و ارزشمندی اطلاعات در حافظه زنبور پیشاهنگ در الگوریتم پیشنهادی را مشخص می‌کند زیرا با توجه به رابطه(۴) :

$$\gamma = \beta^{\left(\frac{d_{max}-d_i}{d_{max}-d_{min}}\right)} \quad (۴)$$

d_i می‌تواند سه مقدار اختیار کند:

$d_i \neq d_{max}$ & d_{min} و $d_i = d_{min}$ ،$d_i = d_{max}$

در حالت اول جمله صورت صفر می‌شود و به تبع آن ضریب γ (تاثیر اطلاعات قبلی) صفر می‌شود، یعنی گرچه این کمان یک بار رؤیت شده (α=۱)، ولی به سبب رؤیت قبلی، برای آن ارزش اضافی یا ضریب افزاینده قرار داده نمی‌شود (γ=۱).

در حالت دوم که $d_i = d_{min}$ است، جمله صورت و مخرج برابر خواهد شد پس γ=β و نشان از آن دارد که گل بعدی که

اطلاعات آن در حال برآورد است، نزدیک‌ترین گلی است که قبلاً هم مشاهده شده است. در حالت سوم که مقداری مابین دو مورد بالا را اختیار می‌کند حالت $1 < \gamma < \beta$ رخ خواهد داد. می‌توان نتیجه گرفت در الگوریتم پیشنهادی γ یکی از شرایط زیر را داراست :

$$d_i = d_{max} \rightarrow \gamma = 1$$
$$d_i = d_{min} \rightarrow \gamma = \beta$$
$$d_i \neq d_{max} \& d_{min} \rightarrow 1 < \gamma < \beta$$

پس یک مسیر دیده شده در نهایت می‌تواند تحت تأثیر ضریبی در بازه $\beta \leq \gamma \leq 1$ قرار گیرد (البته با فرض α=۱) این ضریب β تجربی بوده و بعد از چند تکرار توسط کندو (کاربر) اعمال می‌گردد و با افزایش آن کمان‌های دیده شده در قبل که در حافظه زنبور پیشاهنگ وجود دارند، ارزش بیشتری پیدا می‌کنند و به زنبور پیشاهنگ بیشتر تمایل داده می‌شود تا از حافظه خود پیروی کند و از مسافت دید پیش روی خود (گرچه این تمایل نیز باز احتمالی خواهد بود).

۶-۳ زنبورهای تماشاگر و زنبورهای پیشاهنگ غیر متعهد

زنبورهای پیشاهنگ در یک گام خروج از کندو یک دور نمی‌سازند، بلکه بعد از رؤیت یک یا چند نقطه به کندو بازگردانده می‌شوند و طول مسیر پیموده را به کندو گزارش می‌دهند. در کندو اطلاعات حاصل از زنبورهای پیشاهنگ در معرض زنبورهای دیگری به نام زنبورهای تماشاگر قرار می‌گیرد که این زنبورها وظیفه کمک به زنبور پیشاهنگ و جستجوی اطراف مسیری که او پیموده است را بر عهده دارند. هنگامی که هر زنبور پیشاهنگ به ارائه کیفیت شهد جمع آوری شده (طول مسیر) خود می‌پردازد، زنبورهای تماشاگر با استفاده از مقدار گرد شده Ψ در رابطه(۵) همراه زنبور پیشاهنگ به انتهای آن مسیر حرکت می‌کنند.

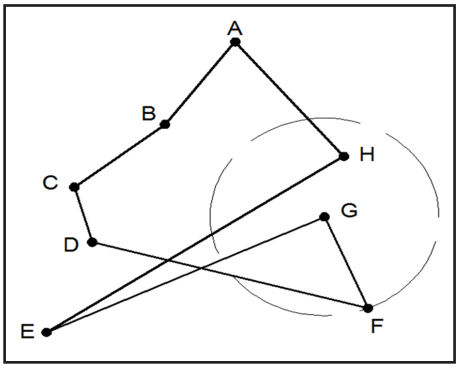
$$\Psi = \frac{\frac{1}{d_k}}{\sum \frac{1}{d_k}} \qquad (۵)$$

در اینجا k نشان دهنده مشخصه (نام) زنبور پیشاهنگی است که مسیر با طول d را پیموده و در حال نمایش آن است. به این طریق هر چه مسیر کوتاه‌تر و با کیفیت‌تر باشد، تعداد بیشتری زنبور تماشاگر به سمت آن حرکت می‌کنند و بیشتر اطراف آن را جستجو می‌کنند. در شرایطی که زنبور پیشاهنگ نتواند هیچ زنبوری را برای مسیر خود به همراه برد، از ادامه مسیر خودداری کرده و آن را رها می‌کند. در این حالت زنبور مذکور یک زنبور پیشاهنگ غیر متعهد نامیده می‌شود که مانند زنبور پیشاهنگی عمل می‌کند که تازه و بدون اینکه نقطه‌ای را در حافظه به عنوان رؤیت شده ثبت داشته باشد از کندو خارج می‌شود و شروع به حرکت می‌کند و به جستجوی دوباره می‌پردازد و به یکباره قسمتی از یک جواب، شامل چند کمان را می‌سازد. اکنون که زنبورهای تماشاگر به همراه زنبورهای پیشاهنگ به مسیرها اعزام شده‌اند، همه آنها بعد از طی چند کمان به کندو باز می‌گردند و مانند زنبورهای پیشاهنگ به ارائه طول مسیر پیموده شده می‌پردازند و روند قبلی دوباره اجرا می‌گردد. این روند تا آنجا ادامه پیدا می‌کند که یک دور کامل ایجاد شود.

۶-۴ بهبود مسیر با الگوریتم در شعاع همسایگی نزدیک

دور تولید شده به وسیله زنبورهای پیشاهنگ به وسیله الگوریتم 2-opt در شعاع همسایگی نزدیک بهبود می‌یابد. الگوریتم 2-opt اصلی، الگوریتمی بسیار کارآ است، اما برای بهبود یک دور در یک گراف با تعداد نقاط N، لازم است (N×N) دور جدید را کنترل کند که این خود فرآیندی زمان‌بَر است. از این رو در انتهای هر دور کامل به بهترین دور انتخاب شده به جای استفاده از الگوریتم 2-opt کلاسیک، از الگوریتم 2-opt اصلاح شده‌ای به نام 2-opt در شعاع همسایگی نزدیک استفاده شد که این اصلاح توسط بنتلی ارائه شده است [Wong and Chong, 2009]. این الگوریتم از همان روش 2-opt استفاده می‌کند تا مسیر را کوتاه‌تر کند اما با محدود و بهینه کردن فضای انتخاب کمان‌های کاندید

برای تعویض با یکدیگر، زمان جستجو که از معایب روش ۲-opt کلاسیک است، کاهش می‌یابد. اساس روش۲-opt اصلاح شده بر پایه یک مشاهده تجربی است که اگر بخواهید تعویضی بین دو کمان ایجاد کنید که منجر به دور کوتاه‌تری شود، باید حداقل طول یک کمان از دو کمان کاندید تعویضی بعد از اعمال ۲-opt کاهش پیدا کند. به عبارت دیگر اگر قرار است یک کمان به عنوان کمانی برای کاهش طول دور کاندید شود، باید فقط با کمان‌هایی تعویض شود که طول آنها، از آن کوتاه‌تر باشد. پس در شعاعی به طول کمان کاندیدی، باید به دنبال نقاطی بگردید که هرگاه کمان مذکور حذف گشته و به آن نقطه وصل شود، حتماً طولش کوتاه‌تر شود. این مفهوم در شکل(۳) با یک مثال تشریح شده است.



شکل ۳. الگوریتم 2-opt در شعاع همسایگی نزدیک

گرافی با هشت نقطه مفروض در شکل ۳ملاحظه می‌شود که از روش کلونی زنبور در آن دوری به صورت A,B,C,D,F,G,E,H,A ایجاد شده است. اگر کمان GF به مرکزیت نقطه G جهت استفاده در الگوریتم 2-opt کاندید شود، باید دایره‌ای به طول آن و حول نقطه G رسم شود، همان طور که ملاحظه می‌شود نقطه H درون این دایره قرار می‌گیرد، پس دو کمان HE و AH برای تعویض با کمان GF مد نظر قرار خواهند گرفت تا با حذف کمان GF و تعویض آنها با هم تشکیل دور جدیدی بدهند.

۶-۵ ذخیره اطلاعات دور حافظه‌ی کندو

در پایان ایجاد هر دور و بهبود آن توسط الگوریتم 2-opt در شعاع همسایگی نزدیک، آن دور با دورهای به دست آمده در تکرارهای قبلی مقایسه می‌شود و اگر دور کوتاه‌تری ایجاد شده بود، دور جدید در حافظه کندو ثبت می‌گردد و اگر دور قبلی کوتاه‌تر بود؛ آن دور همچنان در حافظه باقی می‌ماند و دور جدید تولید شده حذف می‌شود. کمان‌های این دور در آغاز چرخه جدید تولید پاسخ به حافظه زنبورهای پیشاهنگ به عنوان رؤیت شده، القا می‌شود.

۷. اعتبارسنجی الگوریتم پیشنهادی

به منظور اعتبارسنجی و اثبات کارآیی الگوریتم زنبور عسل پیشنهادی به حل چند مسئله معروف پرداخته شده و نتایج حاصل از آن با بهترین نتایج حاصل از روش‌های متداول دیگر (الگوریتم ژنتیک، الگوریتم کلونی مورچه و ...) مقایسه شده و در جدول (۳) نمایش داده شده است. حل با اجرای ۵۰۰ چرخه از الگوریتم پیشنهادی و مقدار $\beta=2$ انجام شده است. به منظور بررسی کاربردی بودن الگوریتم پیشنهادی در مقایسه با نرم‌افزارهای بهینه‌سازی، هر مسئله با استفاده از نرم‌افزار CPLEX نیز حل شد و سپس جوابهای به دست آمده الگوریتم زنبور عسل پیشنهادی با جواب‌های حاصل از حل با استفاده از نرم‌افزار CPLEX مقایسه شد که نتایج آن در جدول ۲ ارائه شده‌اند.

۸. نتیجه‌گیری

با توجه به نتایج به دست آمده می‌توان نتیجه گرفت الگوریتم زنبور عسل پیشنهادی در این تحقیق توانایی حل مسئله مسیریابی خودرو را با متوسط خطا ۰/۳۷۱ ٪ نسبت به سایر روش‌ها دارا است. اصلاح الگوریتم کلونی زنبور عسل به وسیله زنبورهای دیده‌ور پیشنهادی و زنبور پیشاهنگ مقید شده در بهبود جواب‌های به دست آمده از الگوریتم مؤثر بوده است. مقید کردن زنبورهای پیشاهنگ موجب کاهش تقاطع ها گردیده و باعث می‌شود جواب

جدول ۲. مقایسه نتایج به دست آمده*

نام مسأله	تعداد نقاط	بهترین جواب همه روش‌ها (O)	جواب نرم افزار CPLEX (C)	بهترین مقدار به دست آمده با الگوریتم پیشنهادی (B)	درصد خطا $\frac{B-O}{O}$ %	درصد خطا $\frac{B-C}{C}$ %
bayg29	۲۹	۹۰۷۴/۱۴۸	۹۰۴۷/۱۴۸	۹۰۷۴/۱	۰/۰۰	۰/۲۹۸
att48	۴۸	۳۳۵۲۳/۷۰۹	۳۳۵۲۳/۷۰۸۵	۳۳۵۲۳/۷۰۸۵	۰/۰۰	۰/۰۰۰
Eil51	۵۱	۴۲۶	۴۲۸/۸۷۱۸	۴۲۸/۸۷۱۸	۰/۶۷۴	۰/۰۰۰
Berlin52	۵۲	۷۵۴۴	۷۵۴۴/۳۷	۷۵۴۴/۳۶۵۹	۰/۰۰	۰/۰۰۰
St70	۷۰	۶۷۵	۶۷۷/۱۰۹۶	۶۷۷/۱۰۹۶	۰/۳۱۳	۰/۰۰۰
Pr76	۷۶	۱۰۸۱۵۹	۱۰۸۱۵۹/۴۳۸۳	۱۰۸۲۸۰/۴۵۶۶	۰/۱۱۲	۰/۱۱۲
Eil76	۷۶	۵۴۵/۳۸۸	۵۴۴/۳۶۹	۵۵۵/۷۶۸۸	۱/۹۰۳	۲/۰۹۴
gr96	۹۶	۵۱۲/۳۰۹	۵۱۰/۸۸۶	۵۱۲/۶۹۱۲	۰/۰۷۵	۰/۳۵۳
Kroa100	۱۰۰	۲۱۲۸۲	۲۱۲۸۵/۴۴۳۲	۲۱۳۱۱	۰/۱۳۶	۰/۱۲۰
kroaC100	۱۰۰	۲۰۷۵۰/۷۶۳	۲۰۷۵۰/۷۶۲۵	۲۰۸۸۰/۲۰۱۲	۰/۶۲۴	۰/۶۲۴
lin105	۱۰۵	۱۴۳۸۲/۹۹۶	۱۴۳۸۲/۹۹۵۹	۱۴۵۲۹/۵۶۳۲	۱/۰۱۹	۱/۰۱۹
gr120	۱۲۰	۱۶۶۶/۵۰۹	۱۶۱۰/۳۰۷	۱۶۴۹/۳۲۷۸	-۱/۰۳۱	۲/۴۲۳
ch130	۱۳۰	۶۱۱۰/۸۶۱	۶۴۷۲/۲۸۶	۶۱۹۹/۴۰۶۲	۱/۴۴۹	-۴/۲۱۶
ch150	۱۵۰	۶۵۲۸	۶۹۱۵/۹۲۳۳	۶۶۵۹	۲/۰۰۷	-۳/۷۱۵
gr202	۲۰۲	۵۴۹/۹۹۸	۵۰۲/۱۰۰۸	۵۰۱/۱۳۴۷	-۸/۸۸۴	-۰/۱۹۲
Tsp225	۲۲۵	۳۹۱۶	۴۷۷۵/۳۸۸۶	۴۰۴۴/۷۲۰۱	۳,۲۸۷	-۱۵/۳۰۱
A280	۲۸۰	۲۵۷۹	۲۸۲۶/۳۰۷	۲۶۹۷/۴۷۷۱	۴,۵۹۴	-۴/۵۵۸

* اعداد منفی نشانه درصد خطای کمتر از صفر و به معنی برتری جواب الگوریتم پیشنهادی نسبت به جواب مورد مقایسه با آن است.

نهایی کوتاه‌تری حاصل شود. زنبورهای دیده ور با الگوریتم ابداعی موجب شدند که الگوریتم کلونی زنبور عسل با جوابهای بهتری شروع کند و در نهایت به جواب های بهتر و قابل قیاس با سایر الگوریتم ها دست بیابد.

نرم‌افزارهای تجاری مانند CPLEX در ابتدا برتری نامحسوسی از خود نسبت به الگوریتم پیشنهادی نشان می‌دهند، اما با افزایش ابعاد مسئله برتری الگوریتم پیشنهادی نسبت به نرم‌افزار تجاری کاملاً محسوس و نمایان می‌شود و به این علت است که نرم‌افزارهای تجاری برای مصارف تجاری و کوچک بسیار بهینه شده‌اند، اما با پیچیده شدن مسئله راه حل‌های مناسب ارائه نمی‌کنند. این موضوع تأکید مجددی بر موفقیت الگوریتم پیشنهادی در این تحقیق است.

۹. پی‌نوشتها

1. Vehicle Routing Problem
2. Traveling salesman Problem
3. Base Transceiver Station
4. Nondeterministic Polynomial time- Complete
5. Lucic
6. Teodorovic
7. Bhagade
8. Puranik
9. (Nearest Neighbor Algorithm (NNA
10. ji
11. Wu

12. Wong
13. Fragmentation State Transition Rule (FSTR)
14. Fixed Radius Nearest Neighbor 2-opt (FRNN) 2-opt
15. Frequency-Based Pruning Strategy (FBPS)
16. Scout bee
17. Employee bee
18. Onlooker bee

۱۰. مراجع

- Baykasoùlu, A., Özbakır, . L. and Tapkan, P. (2007) "Artificial bee colony algorithm and its application to generalized assignment problem", Vienna, Austria: Itech Education and Publishing,

- Chatterjee, S., Carrera, C. and Lynch, L. A. (1996) "Genetic algorithms and traveling salesman problems", European Journal of Operational Research, Issue 93, pp. 490-510.

- Abu Zitar, R. E. and Shehabat, E. (2007) "Genetic algorithm with solution approach for traveling salesman problem", Neural Network World, 07(5), pp. 497-504.

- Araque, G., J., Kudva, G., Morin, T. and Pekny, J. (1994) "A branch-and-cut algorithm for vehicle routing problems", Annals of Operations Research, Issue 50, pp. 37-59.

- Ascheuer, N., Junger, M. and Reinelt, G. (2000) "A branch and cut algorithm for the asymmetric traveling salesman problem with precedence constraints", Computational optimization and applications, Issue 17, pp. 61–84.

- Barbarosoglu, G. and Ozgur, D. (1999) "A tabu search algorithm for the vehicle routing problem". Computers and Operations Research, Issue 26, pp. 255-270.

- Bel, J. E. and McMullen, P. R. (2004) "Ant colony optimization techniques for the vehicle routing problem", Advanced Engineering Informatics, Issue 18, pp. 41-48.

- Bhagade, A. S. and Puranik, P. V. (2012) "Artificial bee colony (ABC) algorithm for vehicle routing optimization problem", International Journal of Soft Computing and Engineering (IJSCE), 2(2).

- Chiang, W.-C. and Russell, R. A. (1996) "Simulated annealing metaheuristics for the vehicle routing problem with time windows", Annals of Operations Research, Issue 63, pp. 3-27.

- Fiechter, C. (1994) "A parallel tabu search algorithm for large traveling salesman problems", Discrete Applied Mathematics, Issue 51, pp. 243-267.

- Fischetti, M., Salazar Gonzalez, J. J. and Toth, P. (1997) "A branch-and-cut algorithm for the symmetric generalized traveling salesman problem", Operations Research, 3(45), pp. 378-394.

- Geng, X. (2011) "Solving the traveling salesman problem based on an adaptive simulated annealing algorithm with greedy search", Applied Soft Computing, Issue 11, pp. 3680–3689.

- Hasegawa, M., Ikeguchi, T. and Aihara, K. (2002) "Solving large scale traveling salesman problems by chaotic neurodynamics", Neural Networks, Issue 15, pp. 271-283.

- Jemai, J. and Mellouli, K. (2008) "A neural-tabu search heuristic for the real time vehicle routing problem". J Math Model Algor , Issue 7, pp. 161-176.

- Ji, P. and Wu, Y. (2011) "An improved artificial bee colony algorithm for the capacitated vehicle routing problem with time-dependent travel times", The Tenth International Symposium on Operations Research and Its Applications (ISORA 2011), pp. 75-82.

- Li, M., Yi, Z. and Zhu, M. (2009) "Solving TSP by using Lotka–Volterra neural networks", Neurocomputing, Issue 72, pp. 3873–3880.

- Ling, C., Hai-Ying, S. and Shu, W. (2012) "A parallel ant colony algorithm on massively parallel processors and its convergence analysis for the travelling salesman problem", Information Sciences, Issue 199, pp. 31-42.

- Lucic, P. and Teodorovic, D. (2003) "Computing with bees: Attacking complex transportation engineering problems", International Journal on Artificial Intelligence Tools, 3(12), pp. 375-394.

- Ma, J. , Yang, Tao, Hou, Zeng-Gunag, Tan, Min and Liu, Derong (2008) "Neurodynamic programming: a case study of the traveling salesman problem", Neural Comput & Applic, Issue 17, pp. 347-355.

- Malek, M., Guruswamy, M., Pandya, M. and Owens, H. (1989) "Serial and parallel simulated annealing and tabu search algorithms for the traveling salesman problem". Annals of Operations Research, Issue 21, pp. 59-84.

- Manfrin, M., Mauro, B., Stutzle, T. and Dorigo, M. (2006) "Parallel ant colony optimization for the traveling salesman problem". ANTS 2006,LNCS, Volume 4150, pp. 224-234.

- Masutti, T. A. and Castro, L. N. D. (2009) "A self-organizing neural network using ideas from the immune

- Potvin, J.-Y. (1996) “Genetic algorithms for the traveling salesman problem”, Annals of Operations Research, Issue 63, pp. 339-370.

- Wong, L.-P. and Chong, C. S. (2009) ”An efficient bee colony optimization algorithm for traveling salesman problem using frequency-based pruning”, IEEE, pp. 775-783.

- Yonezawa, Y. and Kikuchi, T. (1996) “Ecological algorithm for optimal ordering used by collective honey bee behavior”, 7th International Symposium on Micro Machine and Human, pp. 249-256.

system to solve the traveling salesman problem”, Information Sciences, Issue 179, pp. 1454–1468.

- Moon, C., Kim, J., Choi, G. and Seo, Y. (2002) “An efficient genetic algorithm for the traveling salesman problem with precedence constraints”. European Journal of Operational Research, Issue 140, pp. 606-617.

- Padberg, M. and Rinaldi, G. (1991) “A branch-and-cut algorithm for the resolution of large-scale symmetric traveling salesman”. S.LA.M. Review, Issue 33, pp. 60-100.

پیوست : نمونه داده‌های یکی از مثال‌های حل شده با عنوان Berlin 52

نقطه	X	Y	نقطه	X	Y	نقطه	X	Y	نقطه	X	Y
۱	۶۵۶	۵۷۵	۱۷	۱۴۵	۶۶۵	۳۳	۱۱۵۰	۱۱۶۰	۴۹	۶۰۵	۶۲۵
۲	۲۵	۱۸۵	۱۸	۴۱۵	۶۳۵	۳۴	۷۰۰	۵۸۰	۵۰	۵۹۵	۳۶۰
۳	۳۴۵	۷۵۰	۱۹	۵۱۰	۸۷۵	۳۵	۶۸۵	۵۹۵	۵۱	۱۳۴۰	۷۲۵
۴	۹۴۵	۶۸۵	۲۰	۵۶۰	۳۶۵	۳۶	۶۸۵	۶۱۰	۵۲	۱۷۴۰	۲۴۵
۵	۸۴۵	۶۵۵	۲۱	۳۰۰	۴۶۵	۳۷	۷۷۰	۶۱۰			
۶	۸۸۰	۶۶۰	۲۲	۵۲۰	۵۸۵	۳۸	۷۹۵	۶۴۵			
۷	۲۵	۲۳۰	۲۳	۴۸۰	۴۱۵	۳۹	۷۲۰	۶۳۵			
۸	۵۲۵	۱۰۰۰	۲۴	۸۳۵	۶۲۵	۴۰	۷۶۰	۶۵۰			
۹	۵۸۰	۱۱۷۵	۲۵	۹۷۵	۵۸۰	۴۱	۴۷۵	۹۶۰			
۱۰	۶۵۰	۱۱۳۰	۲۶	۱۲۱۵	۲۴۵	۴۲	۹۵	۲۶۰			
۱۱	۱۶۰۵	۶۲۰	۲۷	۱۳۲۰	۳۱۵	۴۳	۸۷۵	۹۲۰			
۱۲	۱۲۲۰	۵۸۰	۲۸	۱۲۵۰	۴۰۰	۴۴	۷۰۰	۵۰۰			
۱۳	۱۴۶۵	۲۰۰	۲۹	۶۶۰	۱۸۰	۴۵	۵۵۵	۸۱۵			
۱۴	۱۵۳۰	۵	۳۰	۴۱۰	۲۵۰	۴۶	۸۳۰	۴۸۵			
۱۵	۸۴۵	۶۸۰	۳۱	۴۲۰	۵۵۵	۴۷	۱۱۷۰	۶۵			
۱۶	۷۲۵	۳۷۰	۳۲	۵۷۵	۶۶۵	۴۸	۸۳۰	۶۱۰			