

# LA-EO: یک الگوریتم $\tau$ -EO تطبیق پذیر

آیدین خاتم نژاد پاکزاد محمد رضا میبدی

آزمایشگاه سیستم‌های نرم

دانشکده مهندسی کامپیوتر، دانشگاه صنعتی امیر کبیر

تهران ایران

[mmeybodi@aut.ac.ir](mailto:mmeybodi@aut.ac.ir), [khatamnejad@Gmail.com](mailto:khatamnejad@Gmail.com)

**چکیده:** الگوریتم  $\tau$ -EO یک الگوریتم تکاملی است که برای حل مسائل بهینه‌سازی مورد استفاده قرار می‌گیرد. این الگوریتم برای تولید جواب‌های بهتر، در هر مرحله یکی از اجزای آخرین جواب ایجاد شده را انتخاب و مقدار آن را با مقداری جدید جایگزین می‌کند. انتخاب یک جزء از بین اجزای جواب با توجه به شایستگی محلی آنها و یک توزیع احتمال انجام می‌گیرد. این توزیع احتمال که دارای پارامتر  $\tau$  می‌باشد در طول اجرای الگوریتم ثابت بوده و تغییر نمی‌کند؛ به این ترتیب نحوه انتخاب جزء تعویضی در طول اجرای الگوریتم نیز تغییر نکرده و با مساله تطبیق پیدا نمی‌نماید. در این مقاله یک نسخه تطبیق‌پذیر از الگوریتم  $\tau$ -EO به نام  $LA$ -EO معرفی شده است که در آن وظیفه انتخاب جزء تعویضی به یک اتوماتای یادگیر محول شده است. در طول اجرای این الگوریتم، با توجه به شایستگی جواب‌های تولید شده، سیگنال‌های بازخوردی به اتوماتای یادگیر ارسال می‌شود تا به این وسیله نحوه انتخاب جزء تعویضی هرچه بیشتر با مساله تطبیق پیدا کند. به منظور ارزیابی، الگوریتم پیشنهادی بر روی مسائل فروشنده دوره گرد و دو تکه کردن گراف آزمایش گردیده است؛ مقایسه نتایج بدست آمده نشان‌دهنده برتری الگوریتم پیشنهادی بر الگوریتم  $\tau$ -EO دارد.

واژه‌های کلیدی: اتوماتاهای یادگیر، الگوریتم  $\tau$ -EO، بهینه‌سازی حدی، تطبیق‌پذیری، مسایل بهینه‌سازی

## ۱- مقدمه

الگوریتم  $\tau$ -EO [۱] یک الگوریتم تکاملی جدید است که بر اساس ویژگی بحران‌های خود سازمانده [۲][۳] عمل می‌کند. این الگوریتم که برای حل مسائل بهینه‌سازی استفاده شده و جواب‌های مطلوبی ارائه داده است، بر خلاف بسیاری از الگوریتم‌های تکاملی که در هر لحظه از چندین جواب تشکیل شده‌اند تنها از یک جواب تشکیل شده و در هر مرحله آن را بهبود می‌دهد. بهبود جواب در این الگوریتم با انتخاب یکی از اجزای جواب (جزء تعویضی) و جایگزین کردن مقدار آن با مقداری جدید انجام می‌شود. انتخاب جزء تعویضی با توجه به شایستگی اجزا (شایستگی محلی) و یک توزیع احتمال وابسته به پارامتر  $\tau$  انجام می‌شود؛ این انتخاب به شکلی است که اجزا با شایستگی کمتر با احتمال بیشتری انتخاب می‌شوند. از مزایای الگوریتم  $\tau$ -EO می‌توان به داشتن تنها یک پارامتر برای تنظیم و همچنین ساختار بسیار ساده آن اشاره کرد.

از آنجایی که توزیع احتمال استفاده شده در الگوریتم  $\tau$ -EO در طول اجرای الگوریتم ثابت بوده و تغییر نمی‌کند، رفتار الگوریتم برای نمونه‌های<sup>۱</sup> مختلف از یک مساله یکسان بوده و در نتیجه بهترین عملکرد را برای همه آنها ارائه نمی‌کند. برای حل این مشکل در این مقاله نسخه تطبیق‌پذیری از الگوریتم  $\tau$ -EO به نام الگوریتم  $LA$ -EO پیشنهاد شده است؛ این الگوریتم که از یک اتوماتای یادگیر [۴] برای انتخاب جزء تعویضی استفاده می‌کند در طول اجرا با بررسی جواب‌های تولید شده و تولید سیگنال‌های بازخوردی مناسب برای این اتوماتا، با مساله مورد حل تطبیق پیدا می‌کند. نتایج بررسی‌های انجام شده بر روی الگوریتم  $LA$ -EO در حل مسائل فروشنده دوره گرد و دو تکه کردن گراف تاثیر مثبت تطبیق پیدا کردن الگوریتم با مساله را نشان می‌دهد.

ادامه مقاله بدین صورت سازماندهی شده است: در بخش ۲ و ۳ الگوریتم  $\tau$ -EO و اتوماتای یادگیر به اختصار معرفی

<sup>1</sup> Problem Instances

می‌شوند. بخش ۴ به معرفی الگوریتم جدید LA-EO اختصاص دارد. در بخش ۵ مسائل آزمایشی و روش حل آنها توسط دو الگوریتم  $\tau$ -EO و LA-EO شرح داده می‌شوند. در بخش ۶ نتایج حاصل از اجرای الگوریتم‌ها بر روی مسائل انتخاب شده ارائه شده و با هم مقایسه می‌شوند. بخش ۷ نتیجه‌گیری می‌باشد.

## ۲- معرفی $\tau$ -EO

الگوریتم  $\tau$ -EO [۱] یک الگوریتم اکتشافی مبتنی بر پدیده بحران‌های خود سازمانده [۲][۳] است. در این الگوریتم، جواب مساله به آرایه‌ای تبدیل می‌شود که هر یک از عناصر آن معادل یکی از اجزای جواب است. این الگوریتم در هر مرحله (هر تکرار حلقه اصلی) با انتخاب یکی از اجزای جواب (جزء تعویضی) و تغییر مقدار آن جواب را بهبود می‌دهد. در این الگوریتم، برای پیدا کردن جزء تعویضی، ابتدا اجزای جواب به صورت صعودی بر اساس شایستگی محلی آنها مرتب می‌شوند. سپس به هر جزء با توجه به مکان آن در فهرست مرتب شده ( $n$ ) احتمالی متناسب با  $P(n)$  (رابطه ۱) نسبت داده شده و در انتها یکی از اجزا با توجه به این احتمال‌ها انتخاب می‌شود؛ انتخاب جزء تعویضی در این روش به نحوی است که اجزا با شایستگی کمتر با احتمال بالاتری انتخاب می‌شوند. این روش موجب می‌شود تا الگوریتم  $\tau$ -EO بتواند از بسیاری از نقاط بهینه محلی فرار کند.

$$P(n) \sim n^{-\tau} \quad (1)$$

الگوریتم  $\tau$ -EO، برای حل مسائلی نظیر تقسیم گراف به دو زیر گراف، رنگ آمیزی گراف با سه رنگ و فروشنده دوره‌گرد به کار گرفته شده است و جواب‌های مطلوبی ارائه داده است. همچنین نتایج بدست آمده توسط این الگوریتم چه از نظر دقت و چه از نظر سرعت رسیدن به جواب کاملاً قابل رقابت با نتایج بدست آمده توسط الگوریتم‌هایی مانند SA<sup>۲</sup> و الگوریتم ژنتیک نشان داده است.

## ۳- اتوماتای یادگیر<sup>۳</sup>

اتوماتای یادگیر [۴] ماشینی است که می‌تواند تعداد محدودی عمل را انجام دهد؛ هرگاه این ماشین عملی را انتخاب می‌کند، عمل انتخاب شده توسط محیط ارزیابی شده و نتیجه آن به صورت یک سیگنال بازخوردی مثبت (در صورت مناسب بودن عمل) یا منفی (در صورت نامناسب بودن عمل) به اتوماتا بازگردانده می‌شود. مقدار این سیگنال در انتخاب اعمال بعدی تاثیر می‌گذارد. هدف این فرایند این است که اتوماتا بعد از گذشت مدتی به سمت مناسب‌ترین عمل خود در محیط میل کرده و یا به عبارت دیگر یاد می‌گیرد که کدام عمل بهترین عمل است. برای اطلاعات بیشتر درباره اتوماتاهای یادگیر می‌توان به [۵] مراجعه نمود.

## ۴- الگوریتم پیشنهادی

از آنجایی که توزیع احتمال استفاده شده در الگوریتم  $\tau$ -EO در طول اجرای الگوریتم ثابت بوده و تغییر نمی‌کند، رفتار الگوریتم برای نمونه‌های مختلف از یک مساله یکسان بوده و در نتیجه بهترین عملکرد را برای همه آنها ارائه نمی‌کند. برای حل این مشکل در این مقاله نسخه تطبیق‌پذیری از الگوریتم  $\tau$ -EO به نام الگوریتم LA-EO پیشنهاد شده است؛ این الگوریتم که از یک اتوماتای یادگیر برای انتخاب جزء تعویضی استفاده می‌کند در طول اجرا با بررسی جواب‌های تولید شده و تولید سیگنال‌های بازخوردی مناسب برای این اتوماتا، با مساله مورد حل تطبیق پیدا می‌کند.

در الگوریتم LA-EO تعداد اعمال اتوماتای یادگیر برابر تعداد اجزای جواب و هر عمل آن متناظر با یکی از مکان‌های فهرست مرتب شده اجزای جواب است؛ به عبارت دیگر انتخاب عمل  $n$  ام توسط اتوماتای یادگیر، معادل انتخاب جزء قرار گرفته در مکان  $n$  ام این فهرست به عنوان جزء تعویضی می‌باشد. به این ترتیب در هر مرحله از این الگوریتم، بعد از تشکیل شدن فهرست مرتب شده اجزا اتوماتای یادگیر با انتخاب عملی مکان جزء تعویضی در فهرست را مشخص می‌کند. در انتهای هر

<sup>2</sup> Simulated Annealing

<sup>3</sup> Learning Automata

۶. انتخاب یک عمل توسط اتوماتای یادگیر و انتخاب جزء متناظر با آن به عنوان جزء تعویضی

۷. انتخاب مقدار جدید برای جزء تعویضی با توجه به مساله

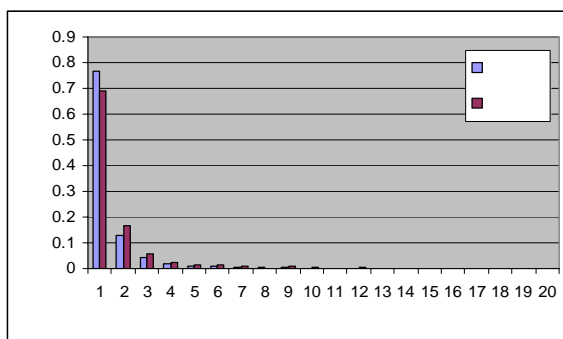
۸. جایگزین کردن مقدار جدید جزء تعویضی و تولید جواب جدید

۹. پاداش دادن و یا جریمه کردن اتوماتای یادگیر با توجه به جواب جدید و جواب‌های قبلی

۱۰. اجرای مرحله ۴ تا توقف الگوریتم

۱۱. برگرداندن بهترین جواب

همانطور که اشاره شد مزیت الگوریتم LA-EO بر الگوریتم  $\tau$ -EO در تطبیق‌پذیری این الگوریتم با نمونه مساله مورد حل است. این الگوریتم ابتدا با بردار احتمالی همانند توزیع احتمال مورد استفاده در الگوریتم  $\tau$ -EO شروع به کار کرده و در طول اجرا با تغییر مقادیر عناصر این بردار، آن را هرچه بیشتر برای حل نمونه مساله مورد نظر تنظیم می‌کند. شکل ۱ وضعیت بردار احتمال اتوماتای یادگیر در الگوریتم LA-EO را در ابتدا و انتهای یک اجرای الگوریتم نمایش می‌دهد.



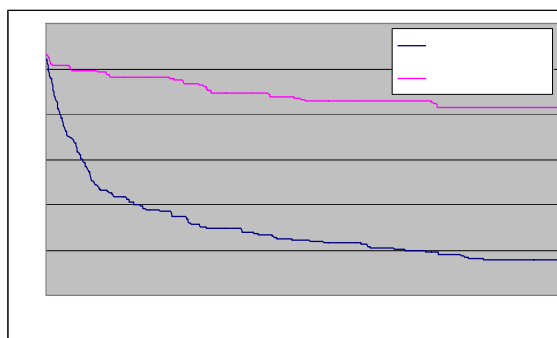
شکل ۱: بردار احتمال اتوماتای یادگیر در الگوریتم LA-EO در ابتدا و انتهای یک اجرای الگوریتم برای ۲۰ عنصر ابتدای بردار

## ۵- مسائل آزمایشی و روش حل آنها

برای ارزیابی کارایی الگوریتم LA-EO و مقایسه آن با روش  $\tau$ -EO از دو مساله بخرنج فروشنده دوره‌گرد<sup>۴</sup> و دو تکه کردن گراف<sup>۵</sup> استفاده شده است. هدف در مساله فروشنده دوره‌گرد پیدا کردن کوتاه‌ترین دور همیلتونی در یک گراف است. نمونه‌هایی از مساله که برای مقایسه

مرحله از الگوریتم LA-EO نیز با توجه به جواب جدید و جواب‌های قبلی، یک سیگنال بازخوردی تولید شده و به اتوماتای یادگیر فرستاده می‌شود تا بردار احتمالات خود را تصحیح کند.

روش انتخاب عمل توسط اتوماتای یادگیر در الگوریتم LA-EO و روش انتخاب مکان جزء تعویضی در الگوریتم  $\tau$ -EO کاملاً یکسان هستند، بنابراین بردار احتمال اتوماتای یادگیر در الگوریتم LA-EO از نظر کارکرد معادل توزیع احتمال در الگوریتم  $\tau$ -EO می‌باشد. آزمایشات نشان داده است که در صورتی که بردار احتمال اتوماتای یادگیر به وسیله توزیع احتمال رابطه ۱ مقداردهی اولیه شود، الگوریتم LA-EO عملکرد بسیار بهتری ارائه می‌دهد؛ نمودار ۱ چگونگی حرکت الگوریتم LA-EO به سمت جواب در حل نمونه‌ای از مساله فروشنده دوره‌گرد را برای حالتی که بردار احتمال اتوماتای یادگیر با توزیع احتمال رابطه ۱ و برای حالتی که این بردار به صورت یکنواخت مقداردهی اولیه شده باشد نمایش می‌دهد.



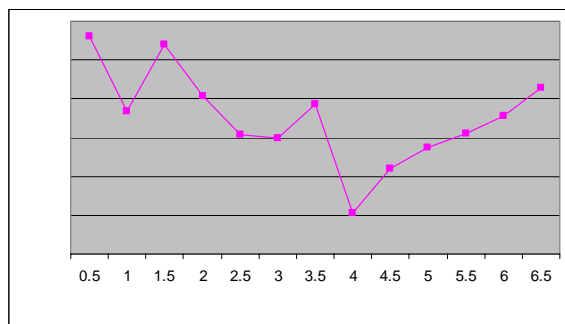
نمودار ۱: مقایسه عملکرد الگوریتم LA-EO بر اساس نحوه مقداردهی اولیه بردار احتمال اتوماتای یادگیر برای دو حالت مقداردهی با توزیع احتمال رابطه ۱ و مقداردهی یکنواخت

با توجه به مطالب شرح داده شده و اعمال تغییرات لازم در الگوریتم  $\tau$ -EO، شکل کلی الگوریتم LA-EO به صورت خواهد بود:

۱. مقداردهی اولیه
۲. مقداردهی بردار احتمال اتوماتای یادگیر
۳. تولید جواب اولیه
۴. محاسبه شایستگی اجزای جواب با توجه به مساله
۵. مرتب کردن اجزای جواب بر اساس شایستگی آنها به صورت صعودی

<sup>۴</sup> Traveling Salesman Problem

<sup>۵</sup> Graph Bi-Partitioning



نمودار ۲: تغییرات عملکرد الگوریتم LA-EO نسبت به پرامتر پاداش برای مساله فروشنده دوره گرد

## ۵-۲- حل مساله دو تکه کردن گراف

برای حل مساله دو تکه کردن گراف توسط الگوریتم LA-EO از روش حل این مساله توسط الگوریتم  $\tau$ -EO در [۹] و [۱۰] استفاده شده است؛ در این روش برای محاسبه شایستگی هر جزء جواب، از نسبت تعداد گره‌های همسایه گره در دو تکه گراف استفاده می‌شود. در پیاده‌سازی انجام شده برای حل مساله دو تکه کردن گراف توسط الگوریتم LA-EO، اتوماتای یادگیر هیچگاه جریمه نشده (پارامتر جریمه همواره برابر صفر است) و در صورتی که جواب تولید شده در هر مرحله بهتر از جواب قبلی الگوریتم بود به عمل انتخاب شده توسط اتوماتای یادگیر پاداش داده می‌شود.

برای محاسبه مقدار پارامتر پاداش، الگوریتم LA-EO به ازای مقادیر مختلف این پارامتر به تعداد ۱۰ مرتبه بر روی مجموعه مسائل آزمایشی مربوط به مساله دو تکه کردن گراف اجرا و مجموع جواب‌های تولید شده به ازای مقادیر مختلف پارامتر محاسبه می‌شود. مقداری از پارامتر پاداش که الگوریتم LA-EO کمترین مجموع جواب را تولید کرده باشد به عنوان مقدار بهینه پارامتر پاداش در نظر گرفته می‌شود. همچنین برای مقایسه الگوریتم‌های LA-EO و  $\tau$ -EO و مشاهده تاثیر تغییرات اعمال شده، مقدار پارامتر  $\tau$  در تمامی اجراها برای ۱,۸۵ که بهترین مقدار این پارامتر برای الگوریتم  $\tau$ -EO می‌باشد قرار داده شده است. نمودار ۳ نشان‌دهنده مجموع جواب‌های تولید شده به ازای مقادیر مختلف پارامتر پاداش می‌باشد. همانطور که مشاهده می‌شود الگوریتم به ازای مقدار ۳ برای پارامتر پاداش کمترین مجموع جواب را تولید کرده است.

عملکرد الگوریتم‌ها استفاده شده‌اند از مرجع [۶] آورده شده‌اند؛ این مجموعه شامل نمونه‌های استاندارد از مساله فروشنده دوره گرد می‌باشند. در مساله دو تکه کردن گراف نیز هدف، تقسیم کردن گراف به دو قسمت است به شکلی که تعداد گره‌های دو قسمت برابر باشد (در صورت فرد بودن تعداد گره‌های گراف، اختلاف تعداد گره‌های دو قسمت برابر ۱ باشد) و ثانیا تعداد یال‌های بین دو قسمت کمترین مقدار ممکن باشد. نمونه‌های استفاده شده برای مقایسه عملکرد الگوریتم‌ها در حل این مساله از مرجع [۷] آورده شده‌اند.

## ۵-۱- حل مساله فروشنده دوره گرد

برای پیاده‌سازی الگوریتم LA-EO برای حل مساله فروشنده دوره گرد از روش حل این مساله توسط الگوریتم  $\tau$ -EO در [۸] استفاده شده است؛ در این روش از فاصله شهرهای با یکدیگر برای محاسبه شایستگی اجزا استفاده است. در پیاده‌سازی انجام شده برای حل مساله فروشنده دوره گرد توسط الگوریتم LA-EO، اتوماتای یادگیر هیچگاه جریمه نشده (پارامتر جریمه همواره برابر صفر است) و در صورتی که جواب تولید شده در هر مرحله بهترین جواب تا آن مرحله باشد به عمل انتخاب شده توسط اتوماتای یادگیر پاداش داده می‌شود.

برای محاسبه مقدار بهینه پارامتر پاداش، الگوریتم به ازای مقادیر مختلف این پارامتر به تعداد ۱۰ مرتبه بر روی مجموعه نمونه مسائل فروشنده دوره گرد اجرا می‌شود. سپس به ازای هر مقدار از پارامتر پاداش، مجموع جواب‌های تولید شده توسط الگوریتم محاسبه شده و مقداری از پارامتر که کمترین مجموع را تولید کرده باشد به عنوان مقدار بهینه این پارامتر در نظر گرفته می‌شود. لازم به ذکر است که برای مقایسه الگوریتم LA-EO و  $\tau$ -EO و مشاهده تاثیر تغییرات اعمال شده، در تمامی این اجراها مقدار پارامتر  $\tau$  برابر ۲,۶ (بهترین مقدار این پارامتر برای الگوریتم  $\tau$ -EO) قرار داده می‌شود. نمودار ۲ مجموع جواب‌های تولید شده به ازای مقادیر مختلف پارامتر پاداش را نمایش می‌دهد. همانطور که مشاهده می‌شود الگوریتم به ازای مقدار ۴ برای این پارامتر کمترین مجموع را ایجاد کرده است.

الگوریتم بر روی مجموعه آزمایشی، در قسمت قبل محاسبه شده است.

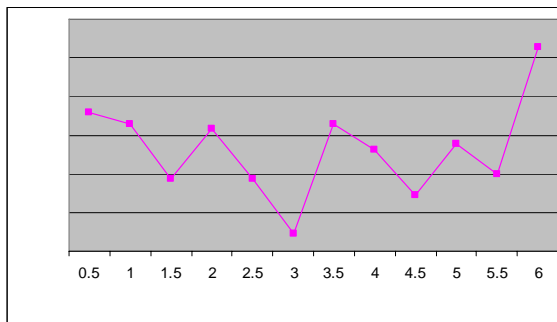
مقایسه الگوریتم‌ها بر اساس مجموعه جواب‌های تولید شده توسط آنها برای نمونه مسائل انتخاب شده در ۱۰ اجرا انجام شده است. نتایج آماری جواب‌های تولید توسط این الگوریتم‌ها در جدول ۲ نمایش داده شده است. همانطور که در این جدول مشاهده می‌شود الگوریتم LA-EO در تقریباً تمامی موارد از نظر میانگین، مقدار بیشینه و مقدار کمینه جواب‌ها از الگوریتم  $\tau$ -EO بهتر می‌باشد.

## ۷- نتیجه گیری

در این مقاله یک نسخه تطبیق‌پذیر از الگوریتم  $\tau$ -EO به نام الگوریتم LA-EO ارائه شده است. در این الگوریتم از یک اتوماتای یادگیر برای انتخاب جزء تعویضی استفاده شده است که در انتهای هر مرحله با توجه به شایستگی جواب تولید شده و جواب‌های قبلی پاداش گرفته و یا جریمه می‌شود و به این ترتیب این الگوریتم در طول اجرا با مساله تطبیق پیدا کند. بررسی انجام شده بر روی الگوریتم LA-EO و مقایسه آن با الگوریتم  $\tau$ -EO نشان می‌دهد که این الگوریتم با احتمال بیشتری نسبت به الگوریتم  $\tau$ -EO جواب‌های مطلوبی ارائه می‌دهد.

## مراجع

- [1] Boettcher, S. and Percus, A. G., "Extremal Optimization: An Evolutionary Local-Search Algorithm", <http://arxiv.org/abs/cs.NE/0209030>
- [2] [http://en.wikipedia.org/wiki/Self-organized\\_criticality](http://en.wikipedia.org/wiki/Self-organized_criticality)
- [3] Per Bak, Chao Tang and Wiesenfeld, K. "Self-organized Criticality", Physical Review A, Vol. 38, No. 1, 1988.
- [4] Narendra K. S. and Thathachar M.A.L., "Learning Automata: An Introduction", Prentice Hall, 1989.
- [5] Thathachar, M. A. L. and Sastry, P. S., "Varieties of Learning Automata: An Overview", IEEE Transaction on Systems, Man, and Cybernetics-Part B: Cybernetics, Vol. 32, No. 6, PP. 711-722, 2002.
- [6] [www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95](http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95)
- [7] <http://staffweb.cms.gre.ac.uk/~c.walshaw/partition>



نمودار ۳: تغییرات عملکرد الگوریتم LA-EO نسبت به پارامتر پاداش برای مساله دو تکه کردن گراف

## ۶- نتایج

برای مقایسه الگوریتم‌ها برای حل مساله فروشنده دوره‌گرد هر کدام از این دو الگوریتم به تعداد ۱۰ مرتبه بر روی نمونه مسائل آزمایشی اجرا و نتایج تولید شده توسط آنها جمع‌آوری شده است. در این مقایسه‌ها، مقدار پارامتر  $\tau$  برای هر دو الگوریتم برابر ۲,۶ در نظر گرفته شده‌اند؛ علت برابر قرار دادن پارامتر  $\tau$  برای این دو الگوریتم بررسی تاثیر تغییرات ایجاد شده در الگوریتم  $\tau$ -EO برای تولید الگوریتم LA-EO است. مقدار پارامتر پاداش الگوریتم LA-EO نیز برابر ۴ در نظر گرفته شده است. نتایج حاصل از اجرای الگوریتم‌ها در جدول ۱ نمایش داده شده است؛ همانطور که مشاهده می‌شود، مقدار میانگین و بیشینه جواب‌های تولید شده توسط الگوریتم LA-EO بر مقدار میانگین و بیشینه جواب‌های تولید شده توسط الگوریتم  $\tau$ -EO برتری دارد در حالی که این وضعیت در مورد مقادیر کمینه بر عکس است. بنابراین می‌توان نتیجه گرفت که در کاربردهایی که از نظر تعداد اجراهای الگوریتم محدودیت وجود دارد (کاربردهای بلادرنگ) استفاده از الگوریتم LA-EO مناسب‌تر است، زیرا با احتمال بالاتری نسبت به الگوریتم  $\tau$ -EO جواب‌های مطلوبی تولید می‌کند.

برای بررسی و مقایسه الگوریتم‌های  $\tau$ -EO و LA-EO در حل مساله دو تکه کردن نیز از نمونه مسائل انتخاب شده در بخش قبل استفاده شده است. برای مشاهده تغییر عملکرد الگوریتم LA-EO نسبت به الگوریتم  $\tau$ -EO مقدار پارامتر  $\tau$  برای هر دو الگوریتم برابر ۱,۸۵ در نظر گرفته شده است. مقدار پارامتر پاداش الگوریتم LA-EO نیز برابر ۳ قرار داده شده است؛ این مقدار با توجه به عملکرد

Threshold", Physics A, vol 32, No.28, pp.5201-5211, 1999.

[10] Boettcher, S. and Percus, A. G., "Extremal Optimization for Graph Partitioning", Physical Review E, vol. 64, pp. 021114, 2001.

[8] Boettcher, S. and Percus, A. G., "Extremal Optimization: Methods derived from Co-Evolution", GECCO-99: Proceedings of the Genetic and Evolutionary Computation Conference, 1999.

[9] Boettcher, S., "Extremal Optimization of Graph Partitioning at the Percolation

نام گراف	میانگین		بیشینه		کمینه	
	LA-EO	$\tau$ -EO	LA-EO	$\tau$ -EO	LA-EO	$\tau$ -EO
A280	2821.8	2860.1	2929	3281	2681	2685
Berlin52	8621.6	8490.6	9437	9257	7963	7888
Burma14	3338.6	3331.5	3371	3472	3323	3323
CH130	7993.6	8032.5	8304	9457	7508	6983
CH150	8777.3	8877.3	9622	10615	7867	7454
EIL101	707.6	711.1	774	791	662	651
EIL76	584.7	580	616	633	563	549
GR202	51133.8	50223.2	54994	54625	47990	46615
GR666	395115.5	400485.4	407600	439334	385765	370356
GR96	61453.7	63278.7	64104	74763	58557	57983
KroA100	27018.5	27775.6	30739	34002	23958	23621
Pcb442	83409.9	92217.9	95164	113634	75697	75533
PR1002	337050.9	340594.6	350690	365298	324912	307220
TSP225	4964.3	4978.3	6129	5810	4445	4269
Ulysses22	7355.2	7275	7599	7646	7126	7014

جدول ۱: مقایسه الگوریتم LA-EO و  $\tau$ -EO در حل مساله فروشنده دوره گرد

نام گراف	میانگین		بیشینه		کمینه	
	LA-EO	$\tau$ -EO	LA-EO	$\tau$ -EO	LA-EO	$\tau$ -EO
3elt	263	273.6	273	292	252	256
4elt	852	901.7	899	952	810	813
Add20	928.2	926.4	988	996	766	784
Add32	73.5	76.2	129	106	40	47
Crack	222.5	223.3	224	230	221	221
CS4	858.3	858.9	866	866	840	851
CTI	1011.5	1046.4	1048	1068	964	1030
Data	260.1	260.1	261	261	252	252
FE_4elt2	306.9	314	314	319	299	307
FE_Sphere	510	510	510	510	510	510
UK	52.1	60.2	56	65	46	57
Whitaker3	136.1	138.5	142	143	135	135
Wing_nodal	3190.4	3249.9	3291	3390	3088	3150

جدول ۲: مقایسه الگوریتم LA-EO و  $\tau$ -EO در حل مساله دو تکه کردن گراف