

یک الگوریتم تقریبی مبتنی بر اتوماتای یادگیر توزیع شده برای حل مسأله مسیریابی وسیله نقلیه با درخواستهای احتمالی

آزمایشگاه سیستم‌های نرم افزاری، دانشکده مهندسی کامپیوتر و فناوری اطلاعات، دانشگاه صنعتی امیرکبیر،
تهران، ایران
mmeybodi@aut.ac.ir

آزمایشگاه سیستم‌های نرم افزاری، دانشکده مهندسی کامپیوتر و فناوری اطلاعات، دانشگاه صنعتی امیرکبیر،
تهران، ایران
aghorbani@aut.ac.ir

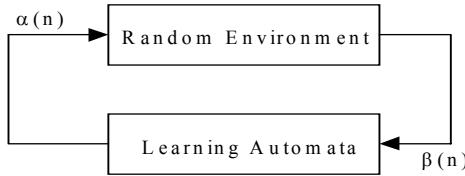
چکیده - مسأله مسیریابی وسیله نقلیه با درخواستهای احتمالی یکی از انواع مسأله مسیریابی وسیله نقلیه میباشد که در آن درخواستهای مشتری ها از قبل مشخص نیست و در زمان ملاقات مشتری مشخص میشود. هدف از حل این مسأله یافتن ترتیبی از مشتری ها می باشد که اگر طبق این ترتیب آنها سرویس دهی شوند متوسط هزینه سفر حداقل گردد. این مسأله یکی از مسائل NP -complete میباشد و بهمین دلیل الگوریتم‌های تقریبی متعددی برای آن طراحی شده است. در این مقاله یک الگوریتم تقریبی مبتنی بر اتوماتای یادگیر توزیع شده برای حل مسأله مسیریابی وسیله نقلیه با درخواستهای احتمالی پیشنهاد میگردد. نتایج آزمایشها نشان داده است که الگوریتم پیشنهادی در مقایسه با الگوریتم‌های موجود نتایج بهتری را تولید کرده است.
کلید واژه - اتوماتاهای یادگیر، اتوماتای یادگیر توزیع شده، الگوریتم تقریبی، مسیریابی وسیله نقلیه با درخواستهای احتمالی

کند که مشتری بعدی را طبق دنباله s سرویس دهد و یا برای بارگیری مجدد به انبار بازگردد. چنانچه تصمیم بر بارگیری مجدد باشد، پس از رفتن به انبار و بارگیری مجدد، سرویس دهی مشتریان طبق دنباله s ادامه پیدا میکند. این عمل تا سرویس دهی کامل به تمام مشتری ها ادامه میابد. بارگیری مجدد وسیله نقلیه پیش از تخلیه کامل بدین دلیل است که مقدار دقیق تقاضای مشتری بعدی از قبل مشخص نیست و لذا ممکن است تقاضای مشتری بعدی بیش از مقدار بار جاری وسیله نقلیه باشد که بارگیری مجدد پیشگیرانه نامیده میشود، باستی انجام گیرد.

هدف از حل مسأله VRPSD یافتن دنباله ای از مشتریان است که کمترین متوسط هزینه را داشته باشد. فرض کنید $s = (0,1,\dots,n)$ یک تور باشد و همچنین فرض کنید پس از سرویس دهی به مشتری j ام بار جاری وسیله نقلیه q باشد، و $f_j(q)$ مشخص کننده متوسط هزینه سفر از گره j تا انتهای تور باشد. بدین ترتیب هزینه متوسط تور برابر با $f_0(Q)$ خواهد بود. اگر L مجموعه مقدار بارهای باشد که وسیله نقلیه پس از اتمام سرویس دهی به مشتری j ام ممکن است داشته باشد، در این صورت $(q_j f_j)$ برای

-۱
مسأله مسیریابی وسیله نقلیه با درخواستهای احتمالی (VRPSD) ببروی گراف کامل $G = (V, A, D)$ تعریف می شود، که $V = \{0, 1, \dots, n\}$ مجموعه گره ها $A = \{(i, j) : i, j \in V, i \neq j\}$ مجموعه یالهای گراف و $D = \{d_{ij} : i, j \in V, i \neq j\}$ ماتریس هزینه میباشد. گره ها انبار و سایر گره ها، مشتری ها می باشند. ماتریس هزینه یک ماتریس متقارن است که قاعده نامساوی مثلثی در آن برقرار میباشد. Q ظرفیت وسیله نقلیه میباشد و فرض بر این است که تقاضای هر مشتری از این ظرفیت بیشتر نمیباشد. تقاضای مشتریان دارای توزیعهای احتمال گسسته $i = 1, \dots, n$ که مستقل از یکدیگر هستند میباشد. $p_{ik} = \Pr(\xi_i = k), k = 1, \dots, K < Q$ احتمال اینکه مشتری i دارای تقاضای k میباشد است.

وسیله نقلیه طبق دنباله $s = \{s(1), s(2), \dots, s(n)\}$ که $s(1)$ انبار میباشد مشتریان را سرویس دهی می کند. سرویس دهی هر مشتری بدین شکل میباشد که پس از مشخص شدن مقدار تقاضای مشتری، وسیله نقلیه تصمیم گیری می



شکل ۱: ارتباط بین اتوماتای یادگیر و محیط

محیط را می توان توسط سه تایی $E \equiv \{\alpha, \beta, c\}$ نشان داد که در آن $\alpha \equiv \{\alpha_1, \alpha_2, \dots, \alpha_r\}$ مجموعه ورودیها، $\beta \equiv \{\beta_1, \beta_2, \dots, \beta_m\}$ مجموعه خروجیها و $c \equiv \{c_1, c_2, \dots, c_r\}$ مجموعه احتمالهای جریمه می باشد. هر گاه β مجموعه دو عضوی باشد، محیط از نوع P می باشد. در چنین محیطی $\beta_1 = 1$ به عنوان جریمه و $\beta_2 = 0$ به عنوان پاداش در نظر گرفته می شود. در محیط از نوع Q ، $\beta(n)$ می تواند به طور گسته یک مقدار از مقادیر محدود در فاصله $[0, 1]$ و در محیط از نوع S، $\beta(n)$ متغیر تصادفی در فاصله $[0, 1]$ است. c_i احتمال اینکه عمل α_i نتیجه نامطلوب داشته باشد می باشد. در محیط ایستا مقادیر c_i بدون تغییر می مانند، حال آنکه در محیط غیر ایستا این مقادیر در طی زمان تغییر می کنند. اتوماتاهای یادگیری به دو گروه با ساختار ثابت و با ساختار متغیر تقسیم بندی میگردند. در ادامه به شرح مختصری درباره اتوماتای یادگیر با ساختار متغیر که در این مقاله از آنها استفاده شده است می پردازیم.

اتوماتای یادگیر با ساختار متغیر : اتوماتای یادگیر با ساختار متغیر توسط ۴ تایی $\{\alpha, \beta, p, T\}$ نشان داده می شود که در آن $\alpha \equiv \{\alpha_1, \alpha_2, \dots, \alpha_r\}$ مجموعه عملهای اتوماتا، $\beta \equiv \{\beta_1, \beta_2, \dots, \beta_m\}$ مجموعه ورودیهای اتوماتا، $p \equiv \{p_1, p_2, \dots, p_r\}$ بردار احتمال انتخاب هر یک از عملهای α و $T[\alpha(n), \beta(n), p(n)] = p(n+1)$ الگوریتم یادگیری می باشد. در این نوع از اتوماتاها، اگر عمل α_i در مرحله n ام انتخاب شود و پاسخ مطلوب از محیط دریافت نماید، احتمال $(n)_i$ افزایش یافته و سایر احتمالها کاهش می یابند. و برای پاسخ نامطلوب احتمال $(n)_i$ کاهش یافته و سایر احتمالها افزایش می یابند. الگوریتم زیر یک نمونه از الگوریتمهای یادگیری خطی برای اتوماتای یادگیر با ساختار متغیر میباشد

$$\text{الف - پاسخ مطلوب} \quad p_i(n+1) = p_i(n) + a[1 - p_i(n)]$$

$$p_j(n+1) = (1 - a)p_j(n) \quad \forall j \quad j \neq i$$

برابر است با: $q \in L_j$

$$f_j(q) = \min \{f_j^P(q), f_j^R(q)\} \quad (1)$$

که

$$f_j^P(q) = d_{j,j+1} + \sum_{k:k \leq q} f_{j+1}(q-k)p_{j+1,k} + \sum_{k:k > q} (b + 2d_{0,j+1} + f_{j+1}(Q-k)p_{j+1,k}) \quad (2)$$

$$f_j^R(q) = d_{j,0} + d_{0,j+1} + \sum_{k=1}^K f_{j+1}(Q-k)p_{j+1,k} \quad (3)$$

شرایط مرزی معادلات فوق از این قرار است:
 $f_j^P(q) = d_{n,0}, q \in L_n$ در معادلات فوق $f_j^P(q)$ متوسط هزینه سفر درصورتی که مستقیماً مشتری بعدی سرویس دهی شود و $f_j^R(q)$ متوسط هزینه درصورتی که بارگیری مجدد پیشگیرانه انجام گیرد میباشد. چنانچه مقدار Q بینهایت باشد مسئله VRPSD به مسئله TSP تبدیل می شود.

تا کنون الگوریتمهای تقریبی متعددی برای حل مسئله مسیریابی وسیله نقلیه و مسیریابی وسیله نقلیه با درخواستهای احتمالی گزارش شده است [5][10-16]. در [5] الگوریتمهایی مبتنی بر کلونی مورچه ها (ACR)، جستجوی تابو (TS)، روش سرد شدن فلزات^۵ و FR^۶ برای این مسئله گزارش شده است. بررسیماس در [19] یک الگوریتم با نام هیوریستیک دوره ای برای حل VRPSD و ینگ در [18]، الگوریتمی برای حل مسئله VRPSD در حالتی که چندین وسیله نقلیه وجود داشته باشد ارائه کرده است. در مقاله مروی [20] الگوریتمهای حل مسئله VRPSD جمع آوری شده است. در [3] توسط قربانی و میدی اولین الگوریتم مبتنی بر اتوماتای یادگیر برای حل مسئله مسیریابی وسیله نقلیه با درخواستهای احتمالی معرفی شده است.

-۲

اتوماتای یادگیر یک مدل انتزاعی است که تعداد محدودی عمل را می تواند انجام دهد. هر عمل انتخاب شده توسط محیطی احتمالی ارزیابی شده و پاسخی به اتوماتای یادگیر داده می شود. اتوماتای یادگیر از این پاسخ استفاده نموده و عمل خود را برای مرحله بعد انتخاب می کند. شکل ۱ ارتباط بین اتوماتای یادگیر و محیط را نشان می دهد.

ابتدا یک اتماتای یادگیر توزیع شده متناظر با گراف مسئله VRPSD ایجاد می شود. اقدامهای هر اتماتای یادگیر انتخاب یکی از مشتری ها می باشد. خروجی اتماتای یادگیر توزیع شده ترتیبی از اقدامها می باشد. با توجه به مطلوب و نامطلوب بودن زیر تور انتخاب شده توسط اتماتاهای یادگیر، اتماتاهای یادگیر در طول این زیر تور پاداش و یا جریمه داده می شود. فرایند ایجاد تور و بروز کردن بردار احتمالات اتماتاهای یادگیر ادامه پیدا می کند تا توری که یک معیار از پیش تعیین شده را ارضا کند پیدا شود.

برای افزایش نرخ همگرایی الگوریتم به خصوص برای گرافهای بزرگ بهتر است تور بدست آمده در انتهای هر مرحله از الگوریتم به طرقی بهبود داده شود. یک راهکار برای انجام این کار استفاده از الگوریتم orOpt [17] و یا orOptTSP [5] می باشد. این الگوریتمها از طریق جابجایی گره ها (مشتری ها) در طول تور بدست سعی در پیدا کردن یک تور بهتر مینماید. فرض کنید s_k گره i تا گره $i+1$ در تور داده شده باشد و پس از حذف s_k از تور، s_k پس از گره j درج میگردد. ابتدا مقدار هزینه ای که با بت حذف s_k کم می شود، محاسبه می گردد و سپس مقدار هزینه ای که با بت درج این رشتہ پس از گره j بدست آمده محاسبه شود. سپس بررسی شود که تور بهتری حاصل گردیده یا خیر. در orOpt از رابطه زیر بدست می آید:

$$\sum_{q=0}^Q [(f_j''(q) - f_j(q)) - (f_i''(q) - f_i(q))] \quad (4)$$

که $f_i(q)$ هزینه سفر از گره i پیش از حذف s_k و $f_i''(q)$ هزینه سفر از گره i تا انتهای تور پس از حذف s_k میباشد. $f_j(q)$ هزینه سفر از گره j پیش از درج s_k و $f_j''(q)$ مقدار هزینه سفر از گره j پس از درج s_k میباشد. در orOptTSP هزینه هر جابجایی از رابطه زیر بدست می آید:

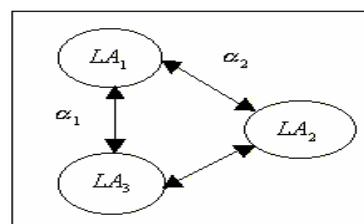
$$d_{i,i+k+1} + d_{j,i+1} + d_{i+k,j+1} - d_{i,i+1} - d_{i+k,i+k+1} - d_{j,j+1} \quad (5)$$

که $d_{i,j}$ فاصله گره های i و j میباشد. در شکل ۳ s_k و

$$p_i(n+1) = (1-b)p_i(n) \quad \text{ب- پاسخ نامطلوب}$$

$p_j(n+1) = \frac{b}{r-1} + (1-b)p_j(n) \quad \forall j \quad j \neq i$
در روابط فوق، پارامتر پاداش و a پارامتر پاداش و b پارامتر جریمه می باشد. با توجه به مقادیر a و b سه حالت را می توان در نظر گرفت. زمانیکه a و b با هم برابر باشند، الگوریتم L_{RP} ¹³ می نامیم. زمانیکه b از a خیلی کوچکتر باشد، الگوریتم L_{REP} ¹⁴ می نامیم. زمانیکه b مساوی صفر باشد، الگوریتم L_{RI} ¹⁵ می نامیم. برای مطالعه بیشتر درباره اتماتاهای یادگیر می توان به [9][8][7][6] مراجعه کرد.

اتوماتای یادگیر توزیع شده (DLA)، شبکه ای از اتماتاهای یادگیر است که برای حل یک مسئله با یکدیگر همکاری می نمایند [2]. تعداد اقدامهای یک اتماتای یادگیر در DLA برابر تعداد اتماتاهای متصل به این اتماتای یادگیر می باشد. انتخاب یک اقدام توسط یک اتماتا در شبکه، اتماتای متناظر با این اقدام را فعال می سازد. عنوان مثال در شکل ۲ اقدام α_2 توسط LA_1 ، اتماتا یادگیر LA_3 را فعال خواهد کرد. اتماتای یادگیر فعال شده (LA_3) سپس یکی از اقدامهای خود را انتخاب می کند که در نتیجه آن یکی از اتماتاهای متصل به آن اتماتا که متناظر با اقدام انتخاب شده می باشد فعال می شود. در هر زمان فقط یک اتماتا در شبکه فعال می باشد. DLA را میتوان توسط گراف $V = \{LA_1, LA_2, \dots, LA_n\}$ که $DLA = (V, E)$ مجموعه $E \subset V \times V$ تعداد اتماتاهای در DLA و n مجموعه لبه های گراف می باشد، تعریف کرد. LA_j زمانی فعال خواهد شد که اقدام j اتماتون k ($k = 1, 2, \dots, n$) LA_k انتخاب شود. تعداد اقدامهای اتماتا (LA_k) برابر درجه خروجی آن گره می باشد. برای اطلاعات بیشتر میتوان به [9] و [21] مراجعه نمود.



شکل ۲ - اتماتای یادگیر توزیع شده (DLA) با ۳ اتماتا یادگیر

می‌گردد.

محل درج آن مشخص شده است.

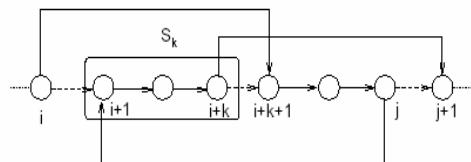
مرحله ۳: دو نقطه i و j در تور بدست آمده بصورت تصادفی انتخاب می‌گردد. سپس در تور جاری دنباله‌ای از گره بین دو گره i و زانتخاب شده و توسط DLA مجدد یک دنباله جدید از این گره‌ها ایجاد می‌شود و جایگزین دنباله قبلی (دنباله‌ای از گره‌های بین i و j) می‌شود. ایجاد دنباله جدید بدین صورت است که ابتدا گره i فعال شده و دنباله جدید با این گره مرتبط باشد. این گره نیز یکی از اعمال مجاز برای آن انتخاب یکی از گره‌های بین i تا $j+1$ می‌باشد. با انتخاب یکی از این گره‌ها اتماتای یادگیر متناظر با گره انتخاب شده فعال می‌شود و تمام اعمال در اتماتاهای دیگر که منجر به انتخاب این گره گردد غیر فعال می‌شود. سپس اتماتای یادگیر متناظر با این گره نیز یکی از اعمال خود را انتخاب می‌کند. انتخاب یک عمل و فعال شدن گره بعدی ادامه پیدا می‌کند تا گره زانتخاب شود و با انتخاب شدن گره زایجاد دنباله مجدد بین i و j توسط DLA تکمیل می‌شود. چنانچه تور حاصل از این تغییر بهتر از تور فعلی باشد، آن تور جایگزین تور فعلی می‌شود و اتماتاهای مربوط به گره‌های بین دو نقطه انتخاب شده، پاداش دریافت می‌کنند.

مرحله ۴:

شبکه کد الگوریتم پیشنهادی در شکل ۴ آورده شده است.

-۵

جهت ارزیابی این الگوریتم از سه نمونه مسئله استفاده شده است. یک مسئله با ۲۰ مشتری و دو مسئله با ۵۰ مشتری. مسئله اول جهت ارزیابی الگوریتم با تعداد گره‌های پایین طراحی شده است و دو نمونه مسئله جهت ارزیابی الگوریتم در تعداد گره‌های بالا مورد استفاده قرار گرفته است که در [۵] معرفی شده است (نمونه ۲: rcn050.f04.d50.r50.s05551 و نمونه ۳: rcn050.f04.d50.r50.s05552)، استفاده می‌شود. برای ارزیابی عملکرد الگوریتم‌ها از هزینه متوسط تور استفاده شده است. در ادامه نمودارهای شبیه‌سازی‌ها آورده شده است. الگوریتم پیشنهادی در این مقاله DLA-ILS نامیده می‌شود و الگوریتم پیشنهادی ارایه شده در [۳] با DLA-VRP نشان داده می‌شود.



شکل ۳: نحوه عمل تابع Opt در این شکل S_k با $k=3$ و محل بعدی آن مشخص شده است.

الگوریتم پیشنهادی با ترکیب DLA و ILS حاصل می‌شود. در الگوریتم ILS ابتدا یک تور اولیه بدست می‌آید سپس با استفاده از یک الگوریتم جستجوی محلی به بهبود جواب بدست آمده می‌پردازد. سپس تور بدست آمده با استفاده از روش 2-exchange تغییر داده می‌شود، چنانچه با این تغییر تور بهتری حاصل گردد، این تور جایگزین تور قبلی می‌شود. استفاده از جستجوی محلی و سپس اعمال 2-exchange ۲-تا حصول شرایط خاتمه الگوریتم ادامه پیدا می‌کند. در الگوریتم ILS تابع 2-exchange می‌کند که بصورت تصادفی دو گره i و j را از تور را انتخاب می‌کند سپس مسیر بین i تا j را معکوس می‌کند. در الگوریتم پیشنهادی به جای استفاده از 2-exchange از DLA برای جابجای گره‌های بین i و j استفاده می‌شود. مراحل اجرای الگوریتم به شرح زیر می‌باشد:

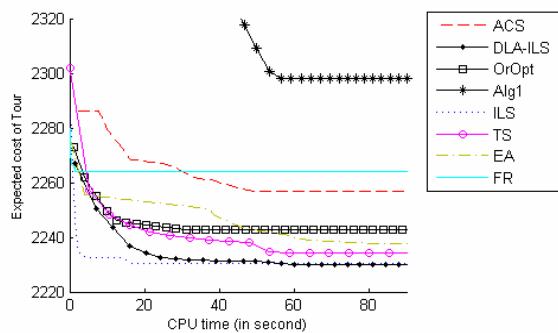
مرحله ۱: یک شبکه از اتماتاهای یادگیر متناظر با گراف مسئله ایجاد می‌شود. هر یک از اتماتاهای یادگیر از الگوریتم یادگیری L_{R-I} استفاده می‌کند. اقدامهای هر اتماتای یادگیر، یالهای خروجی از گره متناظر با ان اتماتای یادگیر می‌باشد. بردار احتمال و اعمال اتماتای یادگیر A_j به ترتیب

$$\alpha^j = \left\{ \alpha_1^j, \alpha_2^j, \dots, \alpha_{r_j}^j \right\} \quad p^j = \left(p_1^j, p_2^j, \dots, p_{r_j}^j \right)$$

می‌باشد. مقدار اولیه اعضای این بردار برابر و مساوی r_j در نظر گرفته می‌شود که r_j تعداد اعمال اتماتای یادگیر A_j می‌باشد. سپس یک تور اولیه که دنباله‌ای از گره‌ها می‌باشد، ایجاد می‌شود.

مرحله ۲: الگوریتم Opt، با جستجو در همسایگی تور بدست آمده سعی می‌کند که تور بهتری را پیدا کند. در صورت پیدا شدن تور بهتر، آن تور جایگزین تور جاری

های بالا در مقایسه با الگوریتمهای موجود نتایج بهتری را تولید کرده است.



نمودار) متوسط هزینه تور حاصل از اجرای الگوریتم‌ها برای مسئله ۳

[1] M. R. Meybodi and H. Beigy, "Solving Stochastic Shortest Path Problem Using Distributed Learning Automata", Proceedings 6th Annual CSI Computer Conference, University of Isfahan, Computer Engineering Department, 2001.

[2] H. Beigy and M. R. Meybodi, "A New Distributed Learning Automata for Solving Stochastic Shortest Path Problem", *Proceedings of the Sixth International Joint Conference on Information Science*, Durham, USA, pp. 339-343, 2002.

[3] A. Ghorbani and M. R. Meybodi, " Utilizing Distributed Learning Automata to Solve Vehicle Routing Problem with Stochastic Demands ", Technical Report, Computer Engineering Department, Amirkabir University, Tehran, Iran, 2007.

[4] M. Alipour and M. R. Meybodi, "A Distributed Learning Automata Based Algorithm for Solving Shortest Path Problem", Technical Report, Computer Engineering Department, Amirkabir University, Tehran, Iran, 2003.

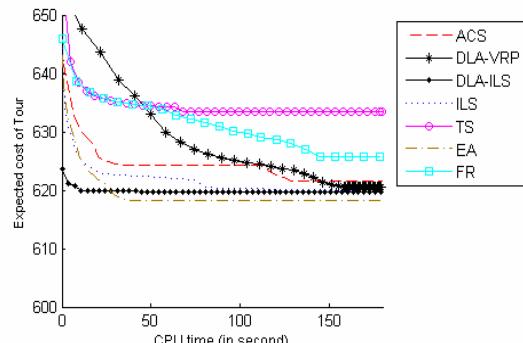
[5] L. Bianchi, M. Birattari, M. Chiarandini, M. Manfrin, M. Mastrolilli, L. Paquete, O. Rossi-Doria, and T. Schiavinotto, "Metaheuristics for the Vehicle Routing Problem with Stochastic Demands", *Lecture Notes in Computer Science*, vol. 3242, pp. 450-460, Springer-Verlag, Heidelberg, Germany, 2004.

[6] K. S. Narendra and K. S. Thathachar, *Learning Automata: An Introduction*, New York: Prentice-Hall, 1989.

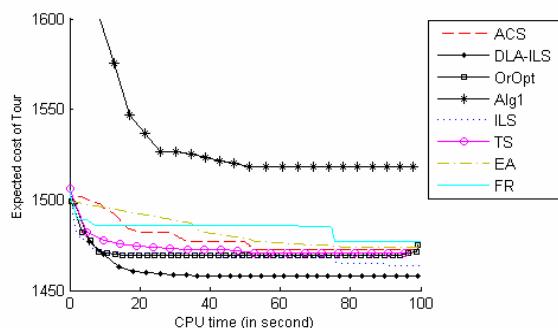
[7] S. Lakshminarayanan, *Learning Algorithms: Theory and Applications*, New York: Springer-Verlag, 1981.

[8] M. R. Meybodi and S. Lakshminarayanan, "On a Class of Learning Algorithms which have Symmetric

الگوریتم پیشنهادی در نمونه اول از تمامی الگوریتم مورد آزمایش بجز الگوریتمهای تکاملی نتایج بهتری را داشته است و برای دو نمونه ۲ و ۳ نتایج بهتری را نسبت به تمام الگوریتمهای آزمایش شده تولید کرده است. الگوریتم DLA-VRP در نمونه‌های با تعداد گره‌های بالا پایین نتایج بهتری را نسبت به الگوریتم‌های کلونی مورچه‌ها، جستجوی تابو و FR دارد و در نمونه‌های با تعداد گره‌های بالا کارایی الگوریتم DLA-VRP کاهش می‌یابد. همچنان که مشاهده می‌شود الگوریتم پیشنهادی در این مقاله در تمام آزمایش‌های صورت گرفته از کارایی مناسبی برخوردار است. با توجه به نمودارها مشاهده می‌شود که دز زمانهای ابتدایی الگوریتمهای موجود جوابهایی بهتری را تولید کردند ولی پس از مدت کوتاهی که یادگیری توسعه اتوماتیک یادگیر حاصل می‌شود، الگوریتم پیشنهادی جوابهایی بهتری را تولید می‌کند.



نمودار) متوسط هزینه تور حاصل از اجرای الگوریتم‌ها برای مسئله ۱



نمودار) متوسط هزینه تور حاصل از اجرای الگوریتم‌ها برای مسئله ۲

-۶

در این مقاله یک الگوریتم تقریبی مبتنی بر اتوماتیک یادگیر توزیع شده برای حل مسئله مسیریابی وسیله نقلیه با درخواستهای احتمالی پیشنهاد گردید. نتایج آزمایشها نشان داد که الگوریتمهای پیشنهادی در نمونه مسائل با تعداد گره

Procedure DLA- ILS

begin

Construct the DLA = (V, E) from graph G as a network of LA

Initialize the probability vector of each automaton

Generate s as a initialSolution

repeat

 $s \leftarrow \text{orOpt}(s)$ // a solution is a sequence of nodes (customers)
 select i and j randomly from tour s; // $i < j$ // perturb nodes between i and j using DLA
 // in a given tour s
 $s1 \leftarrow \text{perturbTourUsingDLA}(s, i, j);$ **if**(cost of $s1 <$ cost of s) **then**
 reward all LA between i and j in tour $s1;$
 $s \leftarrow s1;$ **end if** **until** termination condition not met**end DLA- ILS**

شكل ٤- شبه كد الگوريتم DLA-ILS

¹Vehicle Routing Problem with Stochastic Demands (VRPSD)² Preventive Restocking³ Ant Colony Optimization⁴ Tabu Search⁵ Simulated Annealing⁶ Randomized Farthest Insertion⁷ Bertsimas⁸ Cyclic Heuristic⁹ Yang¹⁰ Stationary¹¹ Non-Stationary¹² Variable Learning Automata¹³ Linear Reward Penalty¹⁴ Linear Reward Epsilon Penalty¹⁵ Linear Reward Inaction¹⁶ Distributed Learning Automata

Behavior under Success and Failure", *Lecture Notes in Statistics*, Berlin: Springer-Verlag, pp. 145-155. 1984.

[9] P. Mars, J. R. Chen and R. Nambir, *Learning Algorithms: Theory and Applications in Signal Processing, Control, and Communication*. CRC Press Inc., 1996.

[10] P. M. Thompson and H. N. Psaraftis. "Cyclic Transfer Algorithms for the Multivehicle Routing and Scheduling Problems", *Operations Research* Vol. 41, pp. 935-946, 1993.

[11] A. Van Breedam, An Analysis of the Behavior of Heuristics for the Vehicle Routing Problem for a Selection of Problems with Vehicle-Related, Customer-Related, and Time-Related Constraints, Ph.D. Dissertation, University of Antwerp. 1994.

[12] M. L. Fisher and R. Jaikumar, "A Generalized Assignment Heuristic for Vehicle Routing", *Networks*, No. 11, pp. 109-124, 1981.

[13] D. M. Ryan, C. Hjorring and F. Glover, "Extensions of the Petal Method for Vehicle Routing", *Journal of the Operational Research Society*, Vol. 44, pp. 289-296, 1993.

[14] J. Renaud and F. F. Boctor, "A Sweep-Based Algorithm for the Fleet Size and Mix Vehicle Routing Problem", *European Journal of Operational Research*, Vol. 140, pp. 618-628, 2002.

[15] É. D. Taillard. "Parallel Iterative Search Methods for Vehicle Routing Problems", *Networks*, No.23, pp. 661-673, 1993.

[16] J. Kelly and J. P. Xu., "A Network Flow-Based Tabu Search Heuristic for the Vehicle Routing Problem", *Transportation Science*, No.30, pp. 379-393, 1996.

[17] I. Or, *Traveling Salesman-type Combinatorial Problems and Their Relation to the Logistics of Blood Banking*, PhD Thesis, Northwestern University, Evanston, IL, 1976.

[18] W. Yang, K. Mathur, and R. H. Ballou, "Stochastic Vehicle Routing Problem with Restocking", *Transportation Science*, Vol. 34, No. 1, pp. 99-112, 2000.

[19] D. J. Bertimas, P. Cervi, and M. Peterson, "Computational Approaches to Stochastic Vehicle Routing Problems", *Transportation Science*, Vol. 29, No. 4, pp. 342-352, 1995.

[20] A. Kenyon and D. P. Morton, "A Survey on Stochastic Location and Routing Problems", *Central European Journal of Operations Research*, Vol.. 9, pp. 277-328, 2002.

[21] H. Beigy and M. R. Meybodi, "Utilizing Distributed Learning Automata to Solve Stochastic Shortest Path Problem", *International Journal of Uncertainty, Fuzziness and Knowledge-based Systems*, Vol. 14 No. 5 pp. 591-615, 2006.