سیستم انبارداری معکوس اتوماتیک: مفاهیم بنیادی، مدل‌سازی و حل مسئلهٔ چینش بهینهٔ اقلام و مسیریابی ربات‌ها

انسیه نیشابوری جامی[1]، الیپس مسیحی[2]*

۱. دانشجوی دکتری مهندسی صنایع، دانشگاه تربیت‌مدرس

۲. استادیار گروه مهندسی صنایع، دانشگاه تربیت‌مدرس

(تاریخ دریافت: ۹۴/۱۱/۱۴، تاریخ دریافت روایت اصلاح‌شده: ۹۵/۰۹/۲۲، تاریخ تصویب: ۹۵/۱۰/۰۴)

چکیده

انبارها و مراکز توزیع، یکی از اجزای مهم در زنجیرهٔ تأمین هستند که مدیریت آن‌ها از اهمیت ویژه‌ای برخوردار است. رویکرد غالب و معمول جمع‌آوری کالا در انبارها بدین‌صورت است که اپراتور به‌صورت پیاده یا با خودرو، اقلام سفارش‌داده‌شده را جمع‌آوری می‌کند. از سال ۲۰۰۶ به بعد، در برخی از انبارهای بزرگ مانند شرکت آمازون- که نقش توزیع‌کننده را دارند- از سیستم جدیدی بهره گرفته شده است که در آن، چندین ربات سیار به زیر قفسه‌های حامل اقلام می‌روند و آن‌ها را به‌سمت اپراتور حرکت می‌دهند. اپراتور نیز اقلام مورد نیاز را از قفسه‌ها برمی‌دارد و سفارش را تکمیل می‌کند. مزایای این سیستم در مقایسه با روش سنتی، افزایش انعطاف‌پذیری، دقت و سرعت آماده‌سازی سفارش‌هاست. از طرفی، این سیستم- که ما آن را «سیستم انبارداری معکوس اتوماتیک (ساما)» می‌نامیم- به‌عنوان راه‌حلی تجاری معرفی شده و تابه‌حال، مستنداتی پژوهشی درمورد آن منتشر نشده است. در این مقاله، سیستم مذکور از دیدگاه مهندسی صنایع مطالعه می‌شود. سپس اجزای آن و ارتباط آن‌ها با یکدیگر تبیین و معرفی می‌شود و دو زیرمسئلهٔ اصلی آن شامل مسائل تخصیص و مسیریابی مطالعه می‌شوند. در ادامه، مدل‌سازی ریاضی مسائل چینش بهینهٔ اقلام درون قفسه‌ها، با هدف مینیمم‌کردن هزینه و همچنین یافتن بهترین مسیر برای حرکت قفسه‌ها ارائه می‌شود. در انتها نیز نتایج حل مسئله با الگوریتم ژنتیک و ماکزیمم جریان ارائه شده است.

واژه‌های کلیدی: الگوریتم ژنتیک، الگوریتم شبیه‌سازی تبرید، دوسطحی، گذاشت و برداشت هم‌زمان، مسئلهٔ مکان‌یابی- مسیریابی.

مقدمه و معرفی سیستم انبارداری معکوس اتوماتیک (ساما)

انبارها اجزایی مهم و اساسی در زنجیرهٔ تأمین هستند و به‌دلیل اهمیتشان، روش‌های مختلفی برای طراحی و مدیریت آن‌ها ارائه شده است. درحال حاضر، انبارداری به شیوه‌های مختلفی صورت می‌گیرد که عمدهٔ آن به‌صورت دستی است و بخشی اندک نیز با بهره‌گیری از فناوری‌های روز مانند سیستم تشخیص فرکانس رادیویی و ربات‌ها و ماشین‌های پیشرفته صورت می‌گیرد. بهره‌گیری از سیستم‌هایی مانند ذخیره و بازیابی اتوماتیک[1] و سیستم تشخیص فرکانس رادیویی[2]، بهبود وضعیت در انبارها و کاهش هزینه‌ها و افزایش کارایی سیستم را به‌دنبال داشته است.

در سال ۲۰۰۶ سیستم نوینی به نام Kiva در صنعت انبارداری شرکت Amazon به‌کار گرفته شد که در آن برخلاف روش‌های موجود، به‌جای اینکه اپراتورها یا تجهیزات به سراغ قفسه‌های حاوی اقلام بروند و اقلام مورد نیاز را جمع‌آوری کنند، به کمک اپراتورها یا ماشین‌آلات (ربات‌های متحرک) به‌سمت ایستگاه‌های جمع‌آوری اقلام و تکمیل سفارش حرکت می‌کنند و پس از برداشت اقلام مورد نیاز، به مکان قبلی خود بازمی‌گردند. همچنین در این سیستم، برای چیدن اقلام درون قفسه‌ها نیز برخلاف روش‌های موجود- که اپراتور یا ماشین‌آلات وظیفهٔ چیدن اقلام را به‌عهده دارند- قفسه‌ها به‌سمت اپراتورها حرکت می‌کنند و پس از چیدن، به مکان اولیه بازمی‌گردند.

متأسفانه جزئیات و مستندات سیستم انبارداری فوق در اختیار جامعهٔ علمی دنیا قرار داده نشد و درحال حاضر، به‌صورت یک راه‌حل صنعتی با نام تجاری Kiva Systems یا Amazon Systems به متقاضیان عرضه می‌شود [۱].

---

* نویسندهٔ مسئول: تلفن: ۰۲۱۸۲۸۸۴۹۳۹، دورنگار: ۰۲۱۸۲۸۸۴۳۹۳ Email: masehian@modares.ac.ir

مقالهٔ حاضر، اولین تلاش برای واکاوی اصول ریاضی این نوع انبارداری است که از این پس با نام «سیستم انبارداری معکوس اتوماتیک (ساما)» معرفی می‌کنیم و به تشریح اجزا و زیرمسائل آن می‌پردازیم.

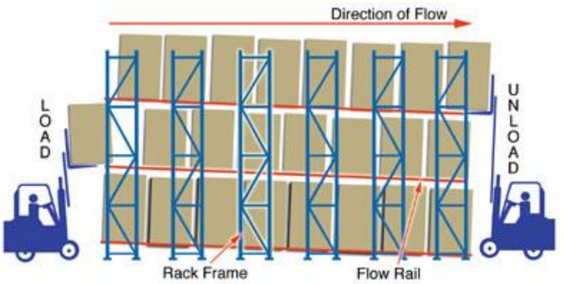
سیستم انبارداری معکوس اتوماتیک، مزیت‌هایی نسبت به روش‌های موجود دارد. یکی از این مزیت‌ها، حذف‌کردن راه‌رفتن اپراتورها در انبارهای سنتی است که با حذف آن، سیستم دو تا سه برابر مؤثرتر می‌شود. همچنین با حرکت موجودی‌ها به‌سمت اپراتور و قرارگرفتن او در جای مشخص، کارایی وی تا دو برابر افزایش می‌یابد. از دیگر مزایای این روش، کاهش خطا در جمع‌آوری سفارش‌هاست که مطابق گزارش‌ها، دقت را در انبارهای مورد استفاده تا ۹۹ درصد افزایش داده است. از دیگر مزایای این روش (حتی در مقایسه با روش‌های اتوماتیک موجود) انعطاف‌پذیری و امکان بسط و توسعهٔ بالای آن است. بدین‌معنا که روش‌های موجود، بعد از نصب و پیاده‌سازی به‌سختی قابل گسترش و جابه‌جایی هستند، اما سیستم انبارداری معکوس، این مشکلات را برطرف ساخته است [۱]. در سیستم انبارداری معکوس اتوماتیک، تعیین تعداد بهینهٔ قفسه‌ها، اپراتورها و رباتهایی که قفسه‌ها را حمل می‌کنند، امری ضروری است. برای بهینه‌سازی چنین سیستمی، زمان‌های بیکاری اپراتورها باید کمینه شود و میزان تجهیزات مورد نیاز بویژه قفسه‌ها و رباتها مینیمم شود.

از تفاوت‌های اصلی ساما با سیستم‌های رایج مانند نوار نقاله‌ها، AGV و پالت- راک‌ها می‌توان به موارد زیر اشاره کرد:

۱. در همهٔ سیستم‌های موجود، قفسه‌ها در مکان ثابت قرار دارند، اما در ساما، قفسه‌ها درحال حرکت‌اند و پس از حرکت، در بهترین مکان قرار می‌گیرند؛

۲. در سیستم‌های کنونی ذخیره‌سازی، یک قلم کالای خاص به‌صورت متمرکز در یک یا چند جای مشخص چیده می‌شود[۳]، اما در ساما، یک قلم کالای خاص را براساس سازگاری بین اقلام می‌توان در چند قفسهٔ مختلف قرار داد؛

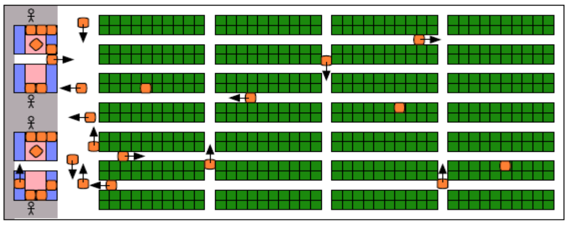
۳. در سیستم‌های اتوماتیک انبارداری، AGVها یا ربات‌ها، اقلام مورد نیاز را از قفسه‌ها برمی‌دارند یا درون آن‌ها قرار می‌دهند و در انبار حرکت می‌کنند. همچنین در سیستم جریان پالت- راک شرکت سیسکو، پالت‌ها به‌وسیلهٔ لیفتراک‌ها در قفسه‌هایی با شیب ملایم گذاشته و از طرف دیگر برداشته می‌شوند (شکل ۱) و حرکت پالت‌ها فقط در درون قفسه (راک) انجام می‌گیرد، اما در ساما، قفسه‌های حامل اقلام به‌وسیلهٔ ربات‌ها در فضای انبار حرکت می‌کنند و اقلام درون آن‌ها به‌وسیلهٔ اپراتور برداشته یا چیده می‌شود.



شکل ۱. سیستم جریان پالت- راک شرکت سیسکو

شایان ذکر است تابع هدف در بیشتر سیستم‌های انبارداری، مینیمم‌کردن هزینه است که در ساما نیز این هدف رعایت شده است.

در شکل ۲ نمای کلی قفسه‌ها، مکان‌های چیدن سفارش و ربات‌های درحال حرکت در راهروهای یک سیستم انبارداری معکوس مشاهده می‌شود. در این شکل، ربات‌های متحرک کوچک و ساده‌ای، قفسه‌های حامل اقلام را به‌سوی ایستگاه‌هایی که اپراتورها قرار گرفته‌اند (سمت چپ شکل)، هدایت می‌کنند.

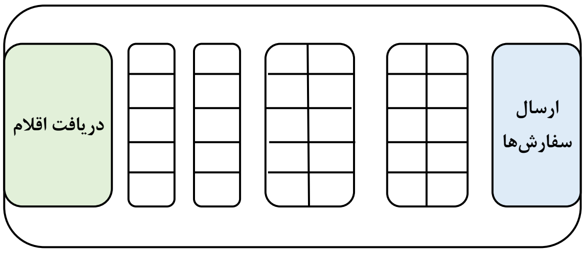


شکل ۲. نمونه‌ای از مکان‌یابی سیستم انبارداری معکوس اتوماتیک [۱]

۱. زیرمسائل اصلی در ساما

در سیستم انبارداری معکوس، سه بخش اصلی وجود دارد: ۱. بخش دریافت اقلام، ۲. انبار اصلی و ۳. بخش ارسال سفارش‌ها. در بخش دریافت اقلام، تعدادی اپراتور وجود دارند که اقلام رسیده به انبار را در قفسه‌ها قرار می‌دهند.

در انبار اصلی، قفسه‌های حاوی اقلام قرار گرفته‌اند. در بخش ارسال سفارش‌ها نیز چند اپراتور سفارش‌ها را تکمیل و برای مشتریان ارسال می‌کنند. در شکل ۳ طرحی از یک نمونه انبار مشاهده می‌شود.



شکل ۳. انبار با بخش دریافت اقلام، انبار اصلی و ارسال سفارش‌ها

ساما را می‌توان از دیدگاه‌های مختلف بررسی کرد. از دیدگاه هوش مصنوعی، همکاری ربات‌ها در این سیستم، به‌صورت چندرباتی[۴] است و به‌دلیل شناخته‌شده‌بودن محیط و کنترل آن، طراحی همکاری آن‌ها ساده است [۱]. در این سیستم، هیچ رباتی در انجام‌دادن کارها به ربات‌های دیگر وابسته نیست. از طرفی همهٔ آن‌ها در تکمیل سفارش مشتری دخالت دارند [۲].

از دیدگاه مهندسی صنایع، طراحی ساما نیازمند بررسی چهار زیرمسئلهٔ اصلی زیر است:

(۱) زیرمسئلهٔ تخصیص، شامل چهار مسئلهٔ تخصیص اقلام واردشده به قفسه‌های منتخب، تخصیص قفسه‌ها به مکان‌های مناسب، تخصیص سفارش‌ها به اپراتورها و تخصیص ربات‌ها به قفسه‌ها برای حمل است.

(۲) زیرمسئلهٔ مسیریابی، یافتن مسیر حرکت قفسه‌ها (ربات‌ها) است؛ به‌گونه‌ای‌که ربات‌ها کوتاه‌ترین مسیر را طی کنند و تداخلی نیز بین آن‌ها اتفاق نیفتد. ربات‌ها پنج وظیفهٔ اصلی را در انبار برعهده دارند که به شرح زیر است:

۱. حرکت ربات از مکان فعلی خود به‌سمت مکان فعلی قفسه؛

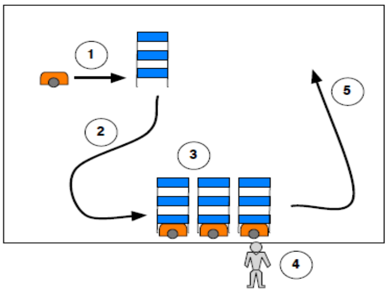
۲. حرکت‌دادن قفسه از مکان فعلی به‌سمت ایستگاه چیدن اقلام یا تکمیل سفارش‌ها؛

۳. ایستادن در صف تا زمانی که نوبت قفسه شود؛

۴. چیدن اقلام روی قفسه یا برداشت موجودی از قفسه توسط اپراتور؛

۵. برگشت قفسه به ناحیهٔ ذخیره‌سازی (بخش اصلی انبار).

در شکل ۴، این وظایف مشاهده می‌شود.



شکل ۴. فعالیت ربات‌ها در سیستم انبارداری معکوس اتوماتیک [۱]

(۳) زیرمسئلهٔ صف، به این دلیل بررسی می‌شود که با تعیین تعداد بهینهٔ اپراتورها و همچنین تخصیص سفارش به آن‌ها، زمان انتظار در صف و همچنین زمان تکمیل سفارش مینیمم شود.

(۴) زیرمسئلهٔ موجودی، تعیین مقدار و زمان مناسب سفارش اقلام است؛ به‌طوری‌که با کمبود موجودی مواجه نشویم.

وظایف و کارهایی را که در انبارها انجام می‌شود، می‌توان از نظر ورود و خروج اقلام به انبار به دو دستهٔ کلی تقسیم کرد: بخشی از فعالیت‌ها مربوط به دریافت اقلام درون انبار و چیدن و ذخیره‌سازی آن‌ها و بخش دیگر مربوط به دریافت سفارش‌ها، تکمیل آن‌ها و ارسال برای مشتریان است. در ادامه، مسائل موجود در این دو حیطه مطرح شده است. در زمان‌های مشخصی، موجودی اقلام قفسه‌ها چک می‌شود و درصورتی‌که نیاز باشد، به تعداد مورد نظر در قفسه‌های منتخب چیده و درصورت نیاز سفارش داده می‌شود.

زیرمسائل مهم در این بخش عبارت‌اند از:

الف) انتخاب قفسه‌های مناسب برای چیدن هر قلم کالا (مسئلهٔ تخصیص)؛

ب) تخصیص نزدیک‌ترین ربات به قفسهٔ مذکور و تعیین مسیر حرکت ربات تا رسیدن به قفسهٔ مذکور (مسئلهٔ تخصیص و مسیریابی)؛

ج) انتخاب ایستگاه مورد نظر برای چیدن هر قلم کالا؛

د) تعداد هر قلم کالا برای چیدن در هر قفسهٔ انتخاب‌شده؛

ه) محل بازگشت هر قفسه با هدف مینیمم فاصله و حمل‌ونقل‌های آتی (مسئلهٔ مسیریابی).

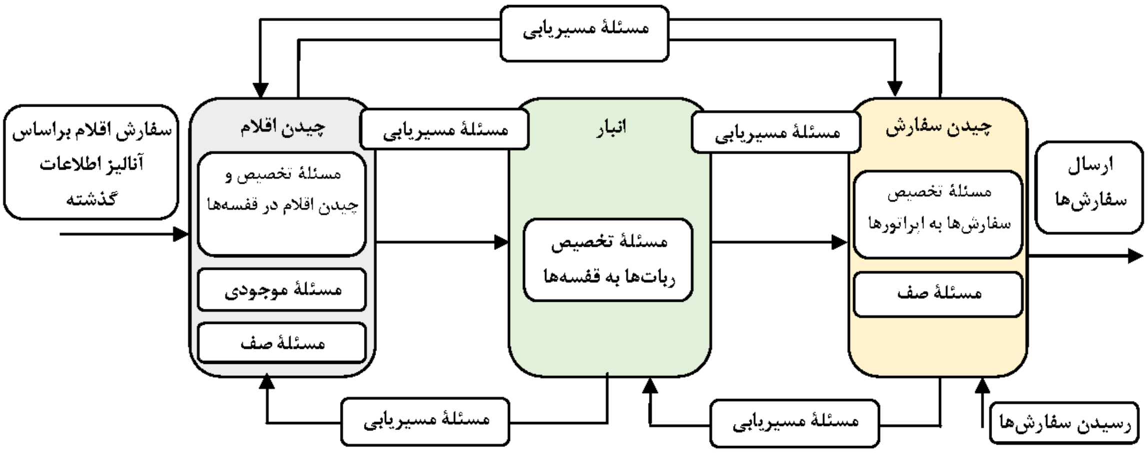
پس از رسیدن سفارش‌ها به انبار برای چیدن به اپراتورها تخصیص داده می‌شود که این تخصیص براساس چند معیار انجام می‌گیرد:

۱. تخصیص سفارش‌ها به ایستگاه‌ها براساس مدت انتظار و تشابه اقلام موجود در سفارش‌ها (مسئلهٔ صف و مسئلهٔ تخصیص)؛
۲. فراخوانی قفسه‌های مورد نیاز به‌وسیلهٔ ایستگاه‌ها؛
۳. تخصیص ربات‌ها به قفسه‌های مورد نظر (مسئلهٔ تخصیص)؛
۴. حرکت قفسه‌ها به‌سمت ایستگاه‌های مورد نظر (مسئلهٔ مسیریابی)؛
۵. تکمیل سفارش‌ها؛
۶. بازگشت قفسه‌ها به جای مناسب (مسئلهٔ مسیریابی)؛
۷. ارسال سفارش‌ها به بخش بسته‌بندی؛
۸. ارسال اطلاعات به بخش‌های دیگر برای به‌روزکردن داده‌ها و اتخاذ تصمیم‌های صحیح.

با توجه به مطالب بالا می‌توان گفت از آنجاکه هر قلم کالا ممکن است در بیش از یک قفسه چیده شده باشد، برای انتخاب قفسهٔ مناسب و تخصیص یک سفارش به یک ایستگاه باید به تشابه اقلام درون سفارش‌های تخصیص‌یافته به یک ایستگاه و اپراتور، مدت انتظار هر سفارش در هر ایستگاه، زمان رسیدن آخرین قفسه و محل بازگشت قفسه‌ها توجه کرد.

این مراحل به‌طور کامل با هم در ارتباطند؛ به‌طوری‌که بر یکدیگر تأثیر متقابل دارند؛ بنابراین، در کل روز، داده‌ها باید در یک سیستم یکپارچه تجزیه و تحلیل شوند. این مسائل در سه بخش اصلی انبار به‌صورت شماتیک در شکل ۵ نشان داده شده‌اند.

در بحث انبارداری، چندین پارامتر مهم مانند ذخیره‌سازی، دسته‌بندی، مسیریابی، اتوماتیک‌بودن سیستم و... بیان می‌شود که بر هزینه‌های سیستم تأثیرگذارند. در پژوهش‌های موجود در این زمینه، اغلب یک یا دو پارامتر بررسی شده است [۳]. در ادامه، با توجه به اهمیت دو موضوع ذخیره‌سازی و مسیریابی، این دو موضوع بررسی شده و مدل ریاضی متناسب با ساختار سیستم انبارداری معکوس برای این دو مسئله بیان شده است.


مسئلهٔ مسیریابی
سفارش اقلام براساس آنالیز اطلاعات گذشته
چیدن اقلام
مسئلهٔ تخصیص و چیدن اقلام در قفسه‌ها
مسئلهٔ موجودی
مسئلهٔ صف
مسئلهٔ مسیریابی
انبار
مسئلهٔ تخصیص ربات‌ها به قفسه‌ها
مسئلهٔ مسیریابی
چیدن سفارش
مسئلهٔ تخصیص سفارش‌ها به اپراتورها
مسئلهٔ صف
ارسال سفارش‌ها
مسئلهٔ مسیریابی
مسئلهٔ مسیریابی
رسیدن سفارش‌ها

شکل ۵. زیرمسائل مطرح در سیستم انبارداری معکوس اتوماتیک (ساما)

بهینه‌سازی سیستم انبارداری معکوس اتوماتیک

از آنجاکه زمان چیدن، بخش جدایی‌ناپذیر زمان تحویل است، کاهش زمان چیدن ممکن است بلافاصله در بهبود خدمات ارائه‌شده به مشتریان نیز تأثیرگذار باشد [۴]؛ بنابراین، کمینه‌کردن زمان چیدن، اهمیتی حیاتی در کنترل فرایند چیدن دارد. برای چیدن یک سفارش خاص، زمان چیدن سفارش برابر زمان یک تور است که در این تور، اقلام مورد نیاز برداشته و سفارش تکمیل می‌شود. این

زمان شامل زمان سفر، زمان جست‌وجو، زمان برداشتن قطعه و زمان برپایی[۵] است. در ساما، زمان سفر مدتی است که طول می‌کشد تا قفسه‌ها به اپراتورها برسند و یک سفارش تکمیل شود. در این سیستم انبارداری، زمان جست‌وجو در عمل ناچیز است، زیرا خود اقلام به‌سمت اپراتور می‌آیند و بقیهٔ زمان‌ها نیز مشابه سیستم‌های دیگر انبارداری است؛ بنابراین، برای بهینه‌کردن ساما باید تمرکز خود را بر زمان سفر قرار دهیم. دو عامل مهم در این زمان، نحوهٔ ذخیره‌سازی اقلام در قفسه‌ها و مسیر حرکت ربات‌هاست که در ادامه به این دو موضوع پرداخته شده است.

۲. مسئلهٔ چیدن بهینهٔ اقلام درون قفسه‌ها

همان‌طورکه ذکر شد، تخصیص اقلام به قفسه‌ها، یکی از مسائل مطرح در سیستم انبارداری معکوس اتوماتیک است که درصورت بهینه‌کردن آن می‌توان سیستم و هزینه‌های آن را بهبود بخشید.

مسئلهٔ واگذاری مکان انبار[۶] نسخهٔ عمومی‌تر مسئلهٔ تخصیص محصولات[۷] است که شامل تخصیص محصولات به قفسه‌های مختلف در انبار است و اغلب هدف آن مینیمم‌کردن هزینه‌های حمل‌ونقل و ماکزیمم‌کردن کاربری فضاست. فرموله‌کردن این مسئله و حل آن نیاز به اطلاعات ورودی دارد که عبارت‌اند از: تعداد، دسترسی، ابعاد و مکان قفسه‌ها، میزان تقاضا، کمیت هر محصول و... [۵].

در این حوزه، پژوهش‌هایی با راه‌حل‌های متفاوت انجام گرفته است که می‌توان به چند مورد از آن‌ها اشاره کرد: موپانی و آدیل [۶]، مدلی را برای تخصیص محصولات دسته‌بندی‌شده به قفسه‌ها طراحی کردند. هیراگو [۷] مدل ریاضی را برای تخصیص اقلام به نواحی انبار ارائه داد و آن را با هیوریستیک سه‌مرحله‌ای حل کرد. اونوت و همکاران [۸] یک مدل ریاضی را برای حل مسئلهٔ PAP در انبارهای چندلایه‌ای معرفی کردند. صانعی و همکاران [۹] یک مدل ریاضی خطی را برای مسئلهٔ PAP با هدف بهینه‌سازی مدیریت موجودی و به‌کارگیری تجهیزات ارائه دادند. گریبرو و همکاران [۵] مقالهٔ [۹] را گسترش دادند و فرض کردند که قفسه‌ها چندطبقه‌اند. در همهٔ مقالاتی که به آن‌ها اشاره شد و پژوهش‌های دیگری که در این حیطه انجام گرفته‌اند، همواره این فرض برقرار بوده است که اپراتورها وظیفهٔ دریافت اقلام را برعهده دارند، اما در مدلی که در ادامه ارائه شده است، فرض می‌شود که قفسه‌ها به‌سمت ایستگاه‌های چیدن حرکت می‌کنند. از طرفی از اطلاعات گذشته برای محاسبهٔ سازگاری بین اقلام استفاده می‌شود.

۳. مسئلهٔ مسیریابی

یکی از مسائل مهم دیگر در انبارداری معکوس که در مدلسازی مسئله هم مدنظر است، مسیریابی است. این مسئله در مقالات مختلفی که در حیطهٔ مسیریابی چیننده در انبار هستند، بررسی شده است. در ادامه، مسائل و مقالات مطرح در این زمینه معرفی می‌شوند.

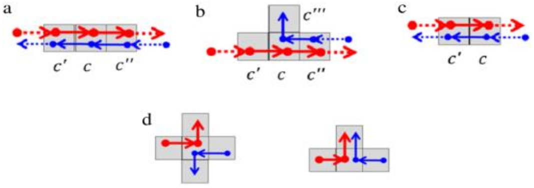
مسئلهٔ فروشندهٔ دوره‌گرد، مشهورترین نوع مسئلهٔ مسیریابی وسیلهٔ نقلیه است. در TSP هدف، یافتن کوتاه‌ترین مسیر برای ویزیت مجموعهٔ شهرها، فروش اجناس و بازگشت به خانه است. TSP جزء مسائل NP-hard است و راه‌حل‌های تقریبی برای حل این مسئله وجود دارد. multi-TSP همان مسئلهٔ فروشندهٔ دوره‌گرد با چند فروشنده است. این مسئله، با روش‌های برنامه‌ریزی عدد صحیح و دیگر روش‌های هیوریستیک قابل‌حل است [۱۰].

تیز و همکاران [۱۱] مسئلهٔ مسیریابی چیدن چند راهرو موازی را بررسی کردند و از آنجاکه مسئلهٔ فروشندهٔ دوره‌گرد، NP-hard به‌شمار می‌رود و تا سه راهرو موازی، با الگوریتم‌های دقیق قابل‌حل است، آن‌ها از هیوریستیک Lin–Kernighan–Helsgaun برای حل بهره گرفتند. مطابق نتایج، استفاده از این الگوریتم سبب کاهش ۴۷ درصد مسیر است. نوروزی و همکاران [۱۲] مدل ریاضی مسیریابی و مکان‌یابی را در زنجیرهٔ تأمین ارائه دادند و با الگوریتم رقابت استعماری حل کردند.

مسئلهٔ مسیریابی وسیلهٔ نقلیه با حمل در بازگشت[۸] با حمل‌ونقل اقلام ارتباط دارد و سریع‌تر از ویزیت‌کردن مکان‌ها به شکل معمولی است. در این مسئله، اقلام به‌وسیلهٔ وسایط نقلیه از دپو یا دپوهای مرکزی به مشتریان تحویل داده می‌شوند. وسایط نقلیه قابلیت آن را دارند که فقط یک سفارش یا ترکیبی از چند سفارش را برداشت کنند و تحویل دهند. در مسئلهٔ VRPB ممکن است برخی محدودیت‌ها مانند ظرفیت، پنجرهٔ زمانی و چند دپو وجود داشته باشد. این مسائل، با روش‌های هیوریستیک و متاهیوریستیک قابل‌حل است [۱۳]. پاندلیس و همکاران [۱۴] مسئلهٔ یافتن مسیر بهینهٔ یک وسیلهٔ نقلیه با شروع از یک دپو و چیدن k محصول برای N مشتری مطابق با

ترتیب مشتریان را بررسی کردند. وسیلهٔ نقلیه اجازه دارد که در حین سفر به دپو برگردد و محصولات جدید را بردارد. محصولات، اندازهٔ یکسان دارند. ستاک و همکاران، مسئلهٔ مسیریابی وسیلهٔ نقلیهٔ چندانباری ظرفیت‌دار را با درنظرگرفتن مسیر بین انبارها درنظر گرفتند و حل کردند [۱۵].

در انبارداری معکوس، ربات‌ها و قفسه‌ها در مسیرهای مشخصی حرکت می‌کنند؛ بنابراین، از روش‌هایی که ارائه شد، نمی‌توان برای حل مسئله استفاده کرد. بدین‌ترتیب، روش‌هایی مطرح می‌شود که برای مسیریابی در مسیرهای مشخص از آن‌ها استفاده می‌شود. داوودی و همکاران [۱۶] روشی برای حرکت دو ربات روی صفحه با تقسیم‌بندی منظم و از روی مسیرهای مشخص ارائه دادند. هدف مقاله، مینیمم‌کردن ماکزیمم طول مسیر است. به‌عبارت دیگر، زمان حرکت آخرین ربات برای رسیدن به هدف را مینیمم می‌کنند. برای حل مسئله، صفحهٔ مشبک را به‌صورت گراف درنظر گرفته‌اند که گره‌ها سلول‌های خالی و بردارها ارتباط میان خانه‌ها را نشان می‌دهد. با تعریف چهار نوع برخورد میان ربات‌ها که در شکل ۶ است، تقدم و تأخر بین دو ربات را در خانه‌های دارای برخورد تعیین می‌کنند و مسئله را حل می‌کنند.

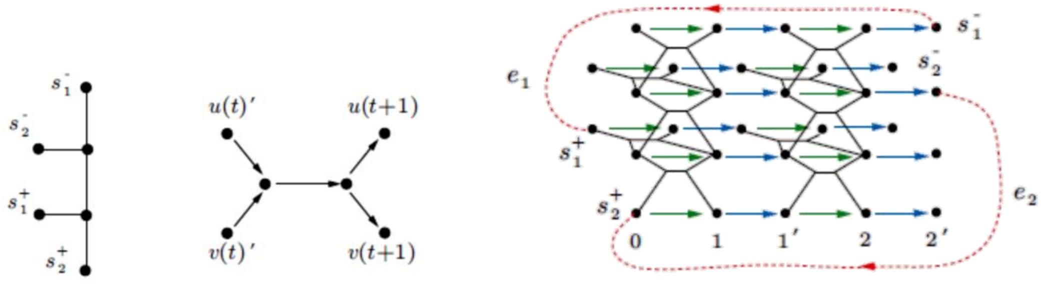


شکل ۶. انواع برخورد بین دو ربات [۱۶]

این روش، برای دو ربات در زمان $O(n)$ قابل‌حل است و برای بیش از دو ربات قابل‌استفاده نیست. یکی از اهداف این پژوهش، گسترش مسئله برای سه ربات است، اما از آنجاکه در مسئلهٔ مورد بررسی، تعداد زیادی ربات وجود دارد، این روش، کارایی زیادی برای حل مسئله نخواهد داشت.

روزبهانی و دآندرآ [۱۷] الگوریتمی برای برنامه‌ریزی مسیر روی شبکهٔ مشبک ارائه دادند. در این تحقیق، فضا به شکل مشبک تقسیم‌بندی شده است و تسهیلات زیادی برای حرکت روی مسیرها برای انجام‌دادن کارهای مشخص وجود دارد. هریک از تسهیلات برای انجام‌دادن کار مورد نظر باید از روی مکان‌های مشخصی عبور کند که این اطلاعات هر لحظه به‌روز می‌شوند. هدف این پژوهش، یافتن مجموعه مسیرهایی است که سرعت متوسط تسهیلات را با درنظرگرفتن امنیت ماکزیمم کنند. جهت حل مسئله، در هر لحظه با در دست داشتن مکان و جهت حرکت هر ربات و با توجه به عدم برخورد بین ربات‌ها، مسیر بهینه را تعیین می‌نماید؛ بنابراین، این روش بیشتر برمبنای داده‌های هندسی و مبانی ریاضی است.

یو و لاوال [۱۸] روشی را مبتنی‌بر تئوری شبکه‌ها ارائه دادند که قابلیت آن را دارد که بدون هیچ محدودیتی، مسیر حرکت ربات‌های زیادی را تعیین کند. مسئلهٔ مسیریابی، چند ربات را منطبق بر یک مسئلهٔ ماکزیمم جریان کرده است؛ به این ترتیب که معادل هر بردار در شبکهٔ اصلی، بردارهایی را به شبکه می‌افزاید که هزینهٔ بردارهای افقی یک و بقیهٔ بردارها صفر است و با جست‌وجوی این شبکه، مسئله را حل می‌کند. در شکل ۷، یک نمونه از گسترش شبکهٔ اولیه و یک نمونه مسئلهٔ حل‌شده به‌وسیلهٔ این روش ارائه شده است.



شکل ۷. شبکهٔ اولیه و شبکهٔ گسترده [۱۸]

مدل‌سازی مسئله

در این بخش، مسئلهٔ مسیریابی ربات‌ها و همچنین ذخیره‌سازی اقلام قفسه‌ها در سیستم انبارداری معکوس، به‌نحوی‌که هزینه‌ها مینیمم شوند، فرموله شده است. در ادامه، مفروضاتی برای ساده‌سازی مدل و تصمیم‌هایی که باید اتخاذ شود، ارائه شده است.

مفروضات

۱. تقاضای مشتریان غیرقطعی است و از توزیع نرمال پیروی می‌کند و تقاضای آن‌ها مستقل از هم است؛

۲. با توجه به تقاضای مشتریان، میزان دریافت هر قلم کالا در بازه‌های زمانی معین مشخص است؛

۳. تعداد اقلام و همچنین تعداد قفسه‌ها ثابت و مشخص است؛

۴. تعداد ربات‌ها را می‌توان در مراحل حل مشخص کرد، اما برای ساده‌سازی مسئله فرض می‌شود که مقدار آن معین است؛

۵. ربات‌ها فقط در مسیرهای ثابت و مشخص حرکت می‌کنند؛

۶. میزان کالای موجود در انبار در هر زمان، از مجموع قفسه‌های انبار، کمتر یا مساوی است؛

۷. جای قفسه‌ها ثابت است و در زمان‌های مختلف، با توجه به اقلامی که روی آن چیده شده است، تغییر نمی‌کند؛

۸. زمان، گسسته فرض شده است.

تصمیم‌ها

۱. **تصمیم‌گیری درمورد تخصیص اقلام به قفسه‌ها:** یکی از تصمیم‌های مهمی که باید در هر زمان اتخاذ شود، این است که هر قلم کالا به کدام قفسه و به چه تعدادی تخصیص یابد و در آن چیده شود.

۲. **تصمیم‌گیری دربارهٔ تخصیص ربات‌ها به قفسه‌ها:** زمانی‌که یک قلم کالا یا یک سفارش می‌رسد، نیاز است که قفسه (قفسه‌های) مورد نظر به‌سمت ایستگاه دریافت اقلام یا ارسال سفارش حرکت کنند که در اینجا، مسئلهٔ مورد نظر تخصیص نزدیک‌ترین ربات به قفسهٔ مورد نظر است.

۳. **تصمیم‌گیری درمورد مسیر حرکت ربات‌ها:** بعد از تخصیص ربات مورد نظر به قفسهٔ مربوط، دو مسئله مطرح است. اولی مسیر حرکت ربات تا رسیدن به قفسهٔ مورد نظر و دومی مسیر حرکت ربات با قفسه تا رسیدن به ایستگاه مورد نظر است؛ به‌طوری‌که ربات‌ها کوتاه‌ترین مسیر را طی کنند و تداخل بین آن‌ها صورت نگیرد.

٤. **تصمیم‌گیری درمورد تعداد بهینهٔ ربات‌ها:** یکی از مسائل مهم با توجه به هزینه‌ها، تعیین تعداد بهینهٔ ربات‌هاست. برای رسیدن به این هدف، در بخش نتایج محاسباتی برای هریک از مسائل با توجه به هزینه‌ها، تعداد بهینهٔ ربات‌ها مشخص شده است. مجموعهٔ اندیس‌ها، پارامترها و متغیرهای تصمیم به شرح جدول ۱ هستند.

تابع هدف و محدودیت‌های مسئله به شرح زیرند:

$$\min z = z_1 + z_2$$

Subject to:

$$z_1 = 2\varepsilon_1 \sum_k \sum_s \sum_l \sum_i \sum_t d_{stk} \cdot x_{islt} + 2\varepsilon_1 \sum_k \sum_s \sum_l \sum_i \sum_t d'_{stk} \cdot x'_{islt} + p_2 \sum_s \sum_i \sum_j \sum_t V_{ijst} - p_1 \sum_s \sum_i \sum_j \sum_t z_{ijst}$$

$$z_2 = 2\varepsilon_2 \Big(\sum_{r=1}^{R} \sum_{d \in S(r)} \sum_{s=1}^{m \cdot w} \sum_{t=1}^{T} y_{rst} \left(|x_{st} - x_{rt}| + |y_{st} - y_{rt}| \right) + R \cdot CR \Big)$$

(۱)

$$\sum_i q_{it} \le mwh, \qquad \forall t$$

$$o_{iT} \le \sum_{t<T} f_{it} - \sum_{t \le T} o_{it}, \qquad \forall t$$

(۲)

$$\sum_s \sum_l y_{islt} = f_{it}, \qquad \forall i, t$$

(۳)

$$x_{islt} \cdot x_{jsht} = z_{ijst}, \qquad \forall i, j \in T_1, h \ne l$$

(۴)

$$x_{islt} \cdot x_{jsht} = 0, \qquad \forall i, j, \forall s, l = h$$

(۵)

$$\sum_s z_{ijst} \ge 1, \qquad \forall t, \forall i, j = 1, \ldots, n \in T_1$$

(۶)

$$x_{islt} \cdot x_{isht} = 1 - U_{ijst}, \qquad \forall i, j \in T_2$$

(۷)

$$U_{ijst} = 0, \qquad \forall i, j \notin T_2$$

(۸)

$$x_{islt} \cdot x_{iuht} = V_{ijst}, \qquad \forall i, j \in T_1, s \ne u$$

(۹)

$$V_{ijst} = 0, \qquad \forall i, j \notin T_1$$

(۱۰)

$$b_{ist} + b_{jst} \le 1 + c_{ij} \qquad \forall i = 1, \ldots, n-1, \forall j = i+1, \ldots, n$$

(۱۱)

$$\sum_s \sum_l y'_{islt} = o_{iT}, \forall i = 1, \ldots n, \forall t = 1, \ldots, T$$

(۱۲)

$$y_{islt} \le M \cdot x_{islt}, \qquad \forall i = 1, \ldots, n, \forall s = 1, \ldots, S, \forall l = 1, \ldots, h, \forall t$$

(۱۳)

$$y'_{islt} \le M \cdot x'_{islt}, \qquad \forall i = 1, \ldots, n, \forall s = 1, \ldots, S, \forall l = 1, \ldots, h, \forall t$$

(۱۴)

$$b_{ist} \geq \frac{1}{h}\sum_{i=1}^{h} x_{islt}, \qquad \forall i=1,\dots,n, \forall s=1,\dots,S, \forall t \tag{۱۵}$$

$$y_{islt} \leq cap_i, \qquad \forall i=1,\dots,n \tag{۱۶}$$

$$y_{rdt}\cdot y_{r'dt}=0, \qquad \forall d,t, \forall r \neq r' \tag{۱۷}$$

$$y_{rst}\cdot y_{rs't}=0, \qquad \forall r=1,\dots,R, \forall s=1,\dots,m\cdot w, \forall t=1,\dots,T \tag{۱۸}$$

$$x_{islt}, y'_{islt} \in \{0,1\}, \qquad \forall i=1,\dots,n, \forall s=1,\dots,S, \forall l=1,\dots,h, \forall t \tag{۱۹}$$

$$b_{ist} \in \{0,1\}, \qquad \forall i=1,\dots,n, \forall s=1,\dots,S, \forall l=1,\dots,h, \forall t \tag{۲۰}$$

$$Z_{islt}, U_{islt}, V_{islt} \in \{0,1\}, \qquad \forall i=1,\dots,n, \forall s=1,\dots,S, \forall l=1,\dots,h, \forall t \tag{۲۱}$$

$$y_{islt}, x'_{islt} \geq 0, \qquad \forall i=1,\dots,n, \forall s=1,\dots,S, \forall l=1,\dots,h, \forall t \tag{۲۲}$$

$$y_{rdt} \in \{0,1\}, \qquad \forall r=1,\dots,R, \forall d \in D, \forall t=1,\dots,T \tag{۲۳}$$

قسمت اول تابع هدف اول، هزینهٔ مسافت طی‌شده به‌وسیلهٔ همهٔ قفسه‌ها را برای اقلام مختلف در زمان‌های مختلف به‌سمت ایستگاه دریافت اقلام با هم جمع می‌کند. قسمت دوم نیز هزینهٔ مسافت طی‌شده به‌وسیلهٔ همهٔ قفسه‌ها را برای اقلام مختلف در زمان‌های مختلف به‌سمت ایستگاه ارسال سفارش با یکدیگر جمع می‌کند. قسمت سوم، مجموع جریمه‌ای را برای اقلامی با $\alpha_{ij} \geq \alpha$ که در یک قفسه قرار ندارند و قسمت چهارم، تخفیفی را برای اقلام با $\alpha_{ij} \geq \alpha$ که در یک قفسه قرار دارند، درنظر می‌گیرد. تابع هدف دوم، مجموع قدم‌های طی‌شده به‌وسیلهٔ ربات‌ها را مینیمم می‌سازد. شایان ذکر است که با استفاده از ضرایب ε_1 و ε_2- که به‌ترتیب هزینهٔ حمل قفسه و ربات برای یک متر مسافت طی‌شده را نشان می‌دهند- جملات توابع هدف، هم‌بعد (دیمانسیون) می‌شوند.

محدودیت (۱) تضمین می‌کند که مجموع تعداد طبقه‌های تخصیص‌داده‌شده به هر قلم کالا در هر زمان کوچک‌تر، مساوی مجموع کل طبقه‌های موجود در انبار است. در این روش، کمبود مجاز نیست؛ بنابراین، محدودیت (۲) بیان می‌کند که میزان سفارش هر قلم کالا در هر زمان باید از مجموع اقلام موجود کمتر باشد. محدودیت (۳) نشان می‌دهد تعداد همهٔ اقلام واردشده در هر زمان، با مجموع تعداد اقلامی که در هر طبقه در آن زمان چیده می‌شود، برابر است. محدودیت (۴) نشان می‌دهد درصورتی‌که زوج اقلام موجود در T_1 در یک قفسه قرار داشته باشند، مقدار Z_{ijst} برابر یک می‌شود. محدودیت (۵) نشان می‌دهد دو قلم کالا نباید در یک طبقه در کنار هم قرار بگیرند. محدودیت (۶) تضمین می‌کند که در هر لحظه، هر زوج قلم موجود در مجموعهٔ T_1 حداقل در یک قفسه قرار گرفته‌اند.

محدودیت (۷) نشان می‌دهد اقلامی که در مجموعهٔ T_2 قرار گرفته‌اند، در یک قفسه نیستند. محدودیت (۸) تضمین می‌کند برای زوج اقلامی که در مجموعهٔ T_2 نیست، مقدار U_{ijst} برابر صفر است. محدودیت (۹) نشان می‌دهد برای اقلامی که در مجموعهٔ T_1 قرار دارند، درصورتی‌که در قفسه‌های جداگانه باشند، مقدار عبارت V_{ijst} یک خواهد شد. محدودیت (۱۰) تضمین می‌کند برای اقلامی که در مجموعهٔ T_1 نیستند، مقدار عبارت V_{ijst} صفر خواهد شد. محدودیت (۱۱) تضمین می‌کند درصورتی‌که دو قلم کالا در یک قفسه قرار گرفته باشند، با هم سازگار هستند ($\alpha_{ij} \geq \alpha$)، اما می‌توانند درصورت سازگاربودن در قفسه‌های جداگانه قرار بگیرند.

محدودیت (۱۲) نشان می‌دهد مجموع اقلامی که در زمان t از روی قفسه‌ها برداشته می‌شود، با میزان سفارش (تقاضای) آن قلم کالا برابر است. محدودیت (۱۳) نشان می‌دهد درصورتی‌که xislt برابر صفر باشد، هیچ قلم کالایی روی s, l قرار نمی‌گیرد و اگر یک باشد، مقدار آن محدود خواهد بود. محدودیت (۱۴) نشان می‌دهد درصورتی‌که x'_{islt} برابر صفر باشد، هیچ قلم کالایی از روی s,l برداشته نمی‌شود و اگر یک باشد، مقدار آن محدود خواهد بود.

براساس محدودیت (۱۵)، اگر b_{ist} یک باشد، حتماً روی یک طبقه در آن قفسه واقع شده است. محدودیت (۱۶) نشان می‌دهد میزان هر قلم کالایی که روی یک طبقهٔ قفسه‌ای چیده می‌شود، نباید از حداکثر ظرفیت آن قلم کالا بیشتر باشد. محدودیت (۱۷) تضمین می‌کند که در یک زمان، دو ربات در یک مکان قرار ندارند. براساس محدودیت (۱۸) نیز ربات در یک زمان فقط می‌تواند یک قفسه را حمل کند و به یک قفسه تخصیص می‌یابد.

جدول ۱. مجموعهٔ اندیس‌ها، پارامترها و متغیرهای تصمیم

اندیس‌ها	
i, j	اقلام
T	دورهٔ زمانی
L	طبقه‌های یک قفسه
S	شمارهٔ قفسه
R	شمارهٔ ربات
D	شمارهٔ مکان

متغیرهای تصمیم	
$= b_{ist}$	۱ اگر قلم i در زمان t در قفسهٔ s باشد ۰ در غیر این صورت
$= y_{islt}$	تعداد اقلام i که به محل s, l در زمان t تخصیص یافته‌اند
$= x'_{islt}$	۱ اگر قلم i از محل s, l در زمان t برداشته شود ۰ در غیر این صورت
$= x_{islt}$	۱ اگر قلم i به محل s, l در زمان t تخصیص یابد یا وجود داشته باشد ۰ در غیر این صورت
$= y'_{islt}$	تعداد اقلام i که از s, l در زمان t برداشته شده‌اند
$= y_{rst}$	۱ اگر ربات r در زمان t به قفسهٔ s تخصیص یابد ۰ در غیر این صورت
$= Z_{ijst}$	۱ اگر اقلام i و j در زمان t در قفسهٔ s باشند و $\alpha_{ij} \geq \alpha$ ۰ در غیر این صورت
$= U_{ijst}$	۱ اگر قلم i در زمان t در قفسهٔ s باشد، اما قلم j نباشد و $\alpha_{ij} < \alpha$ ۰ در غیر این صورت
$= V_{ijst}$	۱ اگر قلم i در زمان t در قفسهٔ s باشد، اما قلم j نباشد و $\alpha_{ij} \geq \alpha$ ۰ در غیر این صورت

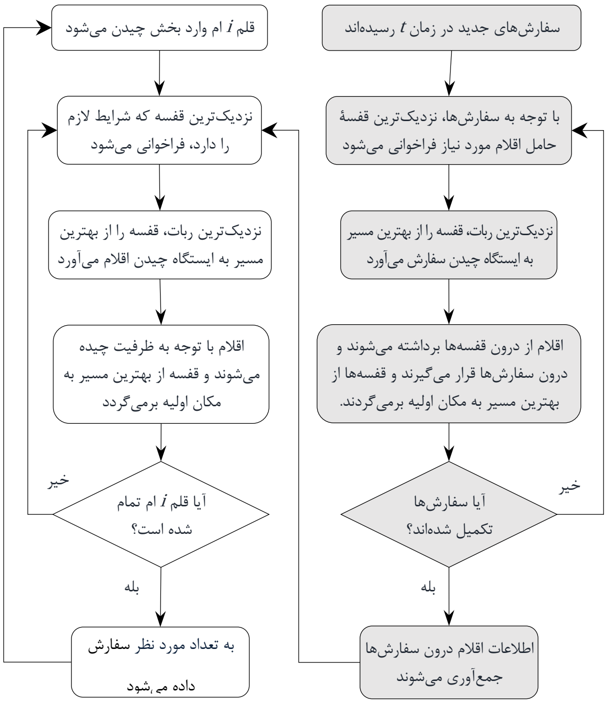
پارامترها	
J	مجموعه نقاطی که هر ربات قادر است با توجه به موانع از آن عبور کند
$S(r)$	مجموعه نقاطی که ربات r ام طی می‌کند و از J انتخاب می‌شود
D	مجموعه نقاطی که ربات‌ها قادرند از روی آن عبور کنند
C	یک ماتریس $N\times N$ که در آن C_{ij} برابر یک است، اگر محصول i با محصول j سازگار باشد؛ در غیر این صورت صفر
P_1	تخفیف، اگر محصول i و j با تواتر بیش از α در سفارش‌ها در یک قفسه قرار بگیرند

پارامترها	
N	تعداد محصولات
M	تعداد قفسه‌های عمودی (برای مثال، در شکل ۳، برابر ۶ است)
W	تعداد قفسه‌های افقی (برای مثال، در شکل ۳، برابر ۵ است)
H	تعداد طبقه‌های هر قفسه
T	دورهٔ زمانی
R	تعداد ربات‌های موجود
P_2	جریمهٔ مورد نظر درصورتی‌که محصول i و j با تواتر بیش از α در سفارش‌ها، در قفسه‌های جداگانه قرار بگیرند
K	مکان‌های دریافت اقلام و سفارش‌ها
D_1	ماتریس فاصله که d'_{slt} فاصلهٔ s, l از مکان دریافت اقلام را در زمان t نشان می‌دهد
D_2	ماتریس فاصله که d'_{slt} فاصلهٔ s, l از مکان ارسال سفارش‌ها را در زمان t نشان می‌دهد
T_1	مجموعهٔ زوج اقلام با $\alpha_{ij} \geq \alpha$
T_2	مجموعهٔ زوج اقلام با $\alpha_{ij} < \alpha$
U	درصد قرارگیری زوج اقلام در سفارش‌های گذشته
F	ماتریس جریان که f_{it} میزان ورود کالای i را در زمان t نشان می‌دهد
α	مقداری بین صفر تا یک که مقیاس قرارگیری اقلام در مجموعه‌های T_1 و T_2 است
O	ماتریس سفارش که O_{it} میزان تقاضای کالای i را در زمان t نشان می‌دهد
$\varepsilon_1, \varepsilon_2$	هزینهٔ حمل قفسه و ربات برای یک متر مسافت طی‌شده
q_{it}	تعداد طبقه‌های اختصاص‌یافته به قلم i ام در زمان t
cap_i	حداکثر تعداد واحدهای قابل‌بارگیری قلم i ام در یک طبقه
α_{ij}	نسبت سفارش قلم i و j ام به کل سفارش‌ها
(s, l)	طبقهٔ l ام از قفسهٔ s ام
(x_{st}, y_{st})	مکان قفسهٔ s ام در زمان t
(x_{rt}, y_{rt})	مکان ربات r ام در زمان t
CR	هزینهٔ خرید هر ربات

نتایج محاسباتی

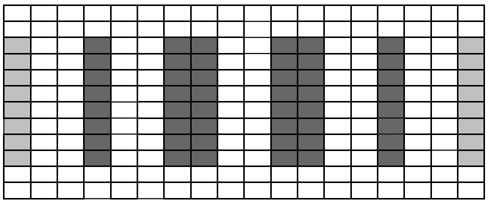
در این تحقیق، مسائل تخصیص و ذخیره‌سازی اقلام درون قفسه‌ها، تخصیص ربات‌ها به قفسه‌ها و مسیریابی ربات‌ها با یکدیگر تلفیق شده‌اند و سیستم بهینه‌ای برای انبار و مدیریت آن ایجاد شده است. برنامه‌ریزی حرکت ربات در مسیرهای خاص انجام می‌شود و سورینک [۱۹] این مسئله را جزء مسائل NP-hard طبقه‌بندی می‌کند. مسئلهٔ تخصیص قفسه‌ها به ربات‌ها، جزء مسائل «ربات‌های تک‌کار- کارهای تک‌ربات- تخصیص گستردهٔ زمانی»[۹] است و این‌ها هم در حیطهٔ مسائل NP-hard هستند [۲۰]؛ بنابراین، به‌منظور حل مسئله لازم است از روش‌های ابتکاری و فراابتکاری استفاده شود. برای حل مسئله از نرم‌افزار متلب استفاده شد. به‌منظور حل مسئلهٔ تخصیص اقلام به قفسه‌ها نیز الگوریتم ژنتیک و برای مسئلهٔ مسیریابی ربات‌ها درصورت اندک‌بودن تعداد ربات‌ها، جست‌وجوی همسایگی‌ها، چک‌کردن تداخل آن‌ها و یافتن مسیر بهینه به‌کار گرفته شد، اما درصورت زیادبودن تعداد ربات‌ها می‌توان از روشی که در مقالهٔ یو و لاوال بیان شده، استفاده کرد. در شکل ۸، نمودار ارتباط میان مسئلهٔ مسیریابی و چیدن اقلام مشاهده می‌شود.

برای حل مسئلهٔ مسیریابی در ابعاد کوچک، الگوریتم از نقطهٔ مبدأ آغاز می‌شود و نزدیک‌ترین گره بعدی به خود را می‌یابد. همین عمل را آن‌قدر تکرار می‌کند تا به نقطهٔ مقصد برسد، اما برای مسئله با ابعاد بزرگ و تعداد ربات‌های بیشتر، گرافی از نقاط موجود برای عبور ربات‌ها تهیه و نقاط مبدأ و مقصد نیز برای هریک مشخص می‌شود. حال شبکه آن‌قدر تکرار می‌شود تا با حل مسئلهٔ ماکزیمم جریان برای شبکهٔ مزبور، کوتاه‌ترین مسیر برای حرکت هر ربات بدون برخورد حاصل شود [۱۸]. در شکل ۹ نحوهٔ شماره‌گذاری مسیرهای مجاز آمده است. در این روش، هریک از نقاط با دو عدد مشخص می‌شوند که عدد اول نشان‌دهندهٔ طول و عدد دوم نشانگر عرض نقطه است. مجموعه نقاطی که ربات‌ها از آن‌ها عبور می‌کنند، در شکل ۸ مشاهده می‌شود.


سفارش‌های جدید در زمان t رسیده‌اند
با توجه به سفارش‌ها، نزدیک‌ترین قفسهٔ حامل اقلام مورد نیاز فراخوانی می‌شود
نزدیک‌ترین ربات، قفسه را از بهترین مسیر به ایستگاه چیدن سفارش می‌آورد
اقلام از درون قفسه‌ها برداشته می‌شوند و درون سفارش‌ها قرار می‌گیرند و قفسه‌ها از بهترین مسیر به مکان اولیه برمی‌گردند.
آیا سفارش‌ها تکمیل شده‌اند؟
خیر
بله
اطلاعات اقلام درون سفارش‌ها جمع‌آوری می‌شوند
قلم i ام وارد بخش چیدن می‌شود
نزدیک‌ترین قفسه که شرایط لازم را دارد، فراخوانی می‌شود
نزدیک‌ترین ربات، قفسه را از بهترین مسیر به ایستگاه چیدن اقلام می‌آورد
اقلام با توجه به ظرفیت چیده می‌شوند و قفسه از بهترین مسیر به مکان اولیه برمی‌گردد
آیا قلم i ام تمام شده است؟
خیر
بله
به تعداد مورد نظر سفارش داده می‌شود

شکل ۸. نمودار مسئلهٔ مسیریابی و چیدن اقلام

$$D=\{(1,1),(2,1),\ldots,(17,1),(2,2),\ldots,(17,11)\}$$



شکل ۹. نحوهٔ شماره‌گذاری مسیرهای قابل‌عبور

اولین مسئلهٔ مهم در استفاده از الگوریتم ژنتیک، انتخاب کروموزوم‌های مناسب است. کروموزوم‌ها براساس دو متغیر اصلی y_{islt} و y'_{islt} انتخاب شده‌اند. مقدار متغیرهای صفر و یک x_{islt} و x'_{islt} براساس مقادیر y_{islt} و y'_{islt} تعیین می‌شود؛ به‌طوری‌که اگر متغیرهای y_{islt} و y'_{islt} بزرگتر از صفر باشند، متغیرهای x_{islt} و x'_{islt} معادل آن‌ها مقدار یک و در غیر اینصورت مقدار صفر اختیار می‌کنند. کروموزوم‌های y_{islt} و y'_{islt} چهاربعدی‌اند. برای نشان‌دادن کروموزوم‌های بیش از دو بعد باید بقیهٔ ابعاد ثابت باشند و دو بعد دیگر نمایش داده شوند. فرض کنید برای یک مسئلهٔ خاص با ۳ قلم کالا $T=3$، $S=3$ و $l=4$ است. کروموزوم‌ها را می‌توان با ثابت نگه‌داشتن دو بعد انتخابی و متغیربودن دو بعد دیگر، به شکل جدول ۲ نشان داد.

جدول ۲. نمایش کروموزوم‌ها

i = 1, t = 1				
l = 4	l = 3	l = 2	l = 1	
۰	۰	۰	۵	S = 1
۱۲	۰	۱۴	۰	S = 2
۰	۰	۰	۰	S = 3

در این جدول، برخی از ابعاد یک کروموزوم نشان داده شده است؛ برای مثال، عدد ۵ به این معناست که از کالای اول در زمان $t=1$، ۵ عدد در طبقهٔ اول قفسهٔ اول وجود دارد.

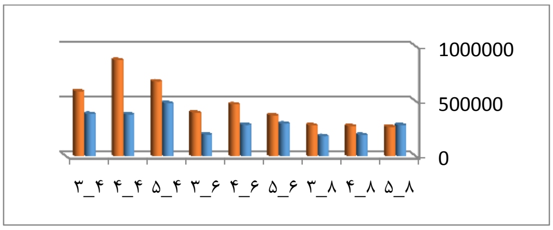
تعداد کروموزوم‌ها بعد از اجراهای مختلف برنامه، ۱۰۰ درنظر گرفته شده است. همچنین تعداد تکرارها ۵۰۰، نرخ جهش ۰٫۰۱، نرخ انتخاب ۷۰ درصد و انتخاب والدین تصادفی است. همچنین ادغام والدها برای تولید نسل بعدی، با یک نقطهٔ تصادفی انجام می‌شود. در جدول ۳، دو کروموزوم بالا والدها و نقطهٔ انتخاب‌شدهٔ تصادفی ۳ است. همچنین دو فرزند ایجادشده در پایین جدول نشان داده شده‌اند. پارامترهای مسئله و همچنین نتایج اجرای برنامه در ادامه ارائه شده است.

$$m=4, w=6, h=3,4,5, cap=100, \varepsilon_1=40,$$
$$p_1=10, p_2=100, N=10,20,50,70,100, \varepsilon_2=100,$$
$$T=10, R=4,6,8,10, CR=1000000$$

جدول ۳. ادغام والدها برای تولید نسل جدید

والد دوم					والد اول				
l=4	l=3	l=2	l=1		l=4	l=3	l=2	l=1	
۱۲	۰	۸	۰		۰	۰	۰	۵	S=1
۰	۰	۰			۱۲	۰	۱۴	۰	S=2
۰	۱۰	۰	۰		۰	۰	۰	۰	S=3
فرزند دوم					فرزند اول				
۰	۰	۸	۰		۱۲	۰	۰	۵	S=1
۱۲	۰	۰	۷		۰	۰	۱۴	۰	S=2
۰	۱۰	۰	۰		۰	۰	۰	۰	S=3

همان‌طورکه از نتایج محاسباتی در جدول ۴ مشخص است، با افزایش تعداد رباتها برای هر مسئله، هزینه‌ها افزایش می‌یابد. اگر محدودیت سرمایه را نیز به محدودیت‌های مسئله اضافه کنیم، با توجه به سرمایهٔ دردسترس می‌توان تعداد رباتهای بهینه را با توجه به هزینه‌ها و همچنین سرمایهٔ دردسترس به‌دست آورد. صرف‌نظر از قیمت رباتها و هزینه‌ای که بابت آن سرمایه‌گذاری می‌شود، می‌توان هزینهٔ هریک از حالتهای فوق را برای هر مسئله به‌دست آورد که نمودار هزینه در هر حالت برای هریک از مسائل مطابق شکل ۱۰ است. در شکل اول، هزینه‌ها بدون درنظرگرفتن هزینهٔ رباتها برای دو مسئلهٔ اول یعنی N=10 (آبی) و N=20 (قرمز) است. برای مسئلهٔ اول، ۸ ربات و ۳ طبقه و برای مسئلهٔ دوم، ۸ ربات و ۵ طبقه مناسب است.

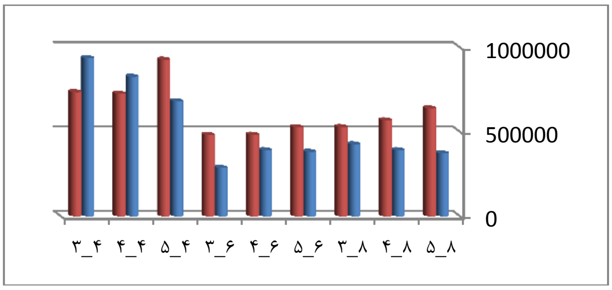


شکل ۱۰. هزینهٔ حالت‌های مختلف رباتها و قفسه‌ها در مسائل با ۱۰ و ۲۰ قلم کالا

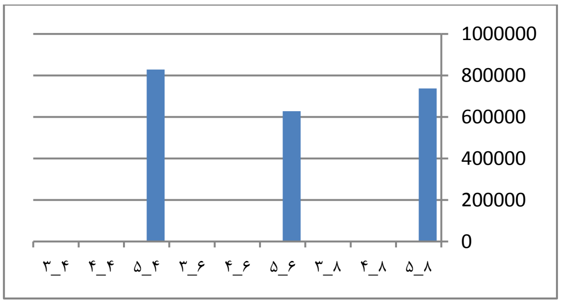
شکل ۱۱ مربوط به هزینه‌های دو مسئلهٔ آبی (N=50) و قرمز (N=70) است. برای مسئلهٔ N=50 بهترین حالت، ۶ ربات و قفسه‌های ۴ طبقه است. برای مسئلهٔ N=70 نیز بهترین حالت ۶ ربات و قفسه‌های ۳ طبقه است.

جدول ۴. نتایج محاسبات

	N	h = ۳		h = ۴		h = ۵	
		هزینه	زمان	هزینه	زمان	هزینه	زمان
R = ۴	۱۰	۴،۳۸۵،۲۷۱	۱۵/۴	۴،۳۸۲،۷۳۷	۱۶,۷	۴،۴۸۳،۷۹۲	۲۸/۹
	۲۰	۴،۵۹۱،۷۲۸	۲۴/۷	۴،۸۷۶،۳۷۲	۲۴,۳	۴،۶۸۱،۷۲۳	۳۷/۵
	۵۰	۴۹،۳۷،۴۳۶	۶۷/۳	۴،۸۲۷،۳۷۲	۸۹,۳	۴،۶۸۲،۶۳۲	۸۵/۶
	۷۰	۴،۷۳۶،۲۷۷	۷۱/۴	۴،۷۲۶،۳۸۲	۹۱,۲	۴،۹۲۸،۶۲۵	۹۲/۵
	۱۰۰	–	–	–	–	۵،۸۲۸،۸۳۲	۹۷/۸
R = ۶	۱۰	۶،۱۹۸،۳۶۲	۱۴/۳	۶،۲۸۷،۳۹۲	۱۶/۴	۶،۲۹۷،۳۶۲	۱۹/۴
	۲۰	۶،۳۹۷،۲۶۲	۱۸/۷	۶،۴۷۲،۶۸۲	۲۳/۵	۶،۳۷۲،۶۷۲	۲۶/۸
	۵۰	۶،۲۸۷،۳۸۳	۶۵/۴	۶،۳۹۲،۸۷۳	۶۳/۲	۶،۳۸۲،۶۳۸	۷۸/۴
	۷۰	۶،۴۸۲،۹۲۲	۷۸/۳	۶،۴۸۳،۷۹۲	۸۲/۷	۶،۵۲۸،۳۶۸	۸۹/۳
	۱۰۰	–	–	–	–	۶،۶۲۸،۲۶۸	۹۳/۷
R = ۸	۱۰	۸،۱۸۳،۲۶۲	۱۵/۷	۸،۱۹۳،۷۷۱	۱۶/۳	۸،۲۸۳،۷۹۲	۱۶/۹
	۲۰	۸،۲۸۲،۳۷۸	۲۸/۴	۸،۲۷۳،۹۳۷	۲۹/۴	۸،۲۶۸،۳۶۹	۳۰/۴
	۵۰	۸،۴۲۷،۳۶۵	۶۷/۲	۸،۳۹۲،۷۳۲	۷۳/۲	۸،۳۷۲،۹۳۷	۷۵/۲
	۷۰	۸،۵۲۹،۷۳۳	۷۵/۸	۸،۵۶۹،۳۷۳	۸۴/۶	۸،۶۳۹،۷۴۳	۸۹/۴
	۱۰۰	–	–	–	–	۸،۷۳۷،۶۴۳	۹۷/۴



شکل ۱۱. هزینهٔ حالت‌های مختلف ربات‌ها و قفسه‌ها برای مسائل با ۵۰ و ۷۰ قلم کالا



شکل ۱۲. هزینهٔ حالات مختلف ربات‌ها و قفسه‌ها برای مسائل با ۱۰۰ قلم کالا

شکل ۱۲ مربوط به هزینه‌های N=100 است که در آن، کمترین هزینه به ۶ ربات و قفسه‌های ۵ طبقه اختصاص دارد. همان‌طورکه در جدول ۴ و شکل ۱۲ مشاهده می‌شود، از آنجاکه تعداد اقلام ۱۰۰ است و براساس فرضیات مسئله، تعداد طبقات در انبار باید بزرگ‌تر یا مساوی تعداد اقلام باشد، این مسئله فقط برای قفسه‌های ۵ طبقه قابل‌حل است.

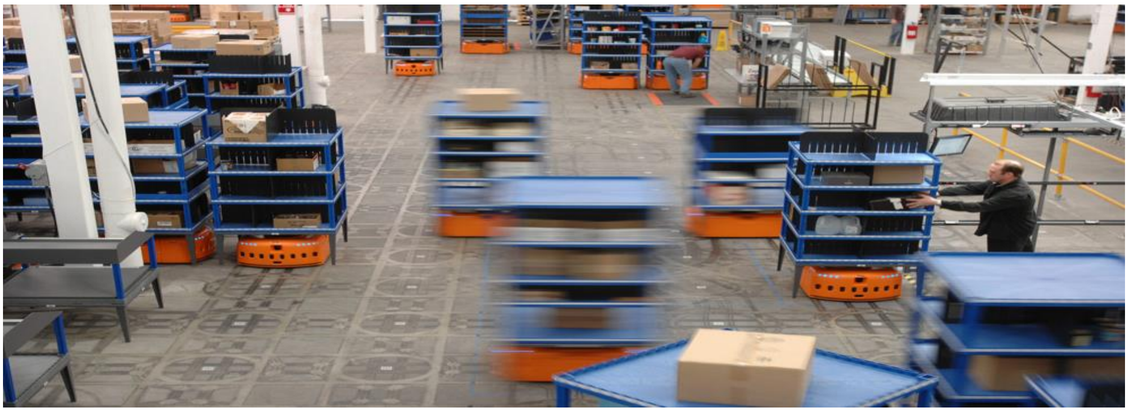
نتیجه‌گیری و پژوهش‌های آتی

در این مقاله، شیوهٔ نوین انبارداری به نام Kiva Systems که در سال‌های اخیر به‌صورت تجاری و گسترده در شرکت Amazon به‌کار رفته است، بررسی و چهار زیرمسئلهٔ اصلی مهندسی آن و نیز ارتباط این زیرمسائل با یکدیگر شناسایی شد.

از آنجاکه سیستم انبارداری مورد مطالعه، ویژگی‌های جدید و منحصربه‌فردی دارد که آن را از انبارهای اتوماتیک با حرکت اقلام به‌وسیلهٔ لیفتراک و AGV و نوار نقاله متمایز می‌سازد، نام «سیستم انبارداری معکوس اتوماتیک (ساما)» به آن اطلاق شد که با الگوبرداری از حرکت خودکار قفسه‌های اقلام به‌سمت اپراتور به‌وسیلهٔ ربات (شکل ۱۳) درجهت معکوس حرکت سنتی جمع‌آوری اقلام به‌وسیلهٔ اپراتور است.

در ادامه، به دو زیرمسئلهٔ مسیریابی ربات‌ها و ذخیره‌سازی (چینش) بهینهٔ اقلام در ساما پرداخته شد و مدل ریاضی این دو زیرمسئله ارائه شد. ویژگی این مدل ریاضی در مقایسه با مدل‌های موجود، توجه به مسیر حرکت ربات‌ها بر مسیرهای مشخص و لحاظ‌کردن درصد اقلام سفارش‌داده‌شده با هم در نحوهٔ چیدن اقلام در قفسه‌ها و همچنین تکمیل سفارش‌هاست. با توجه به NP-hard بودن مدل ریاضی، روش الگوریتم ژنتیک برای حل آن ارائه شد و نتایج، گزارش و تحلیل شدند.

به‌منظور ادامهٔ کار و گسترش این مسئله می‌توان بحث نحوهٔ تخصیص سفارش‌ها به اپراتورها را برای تکمیل به مدل بالا اضافه کرد؛ به‌طوری‌که سفارش‌ها با توجه به تشابهات موجود در اقلام درخواستی دسته‌بندی شوند و برای تکمیل، به اپراتورها اختصاص یابند.



شکل ۱۳. نمایی از به‌کارگیری سیستم Kiva در شرکت آمازون [۱۹]

مراجع

1. Wurman, P.R., D’Andrea, R. and Mountz, M. (2008). “Coordinating hundreds of cooperative, autonomous vehicles in warehouses”, *AI Magazine*, Vol. 29, No. 1, PP. 9–20.
2. Lesser, V.R. (1999). “Cooperative multiagent systems: A personal view of the state of the art”, *IEEE Transactions on Knowledge and Data Engineering*,Vol. 11, No. 1, PP. 133–142.
3. De Koster, R., Le-Duc, T. and Roodbergen, K.J. (2007). “Design and control of warehouse order picking: A literature review”, *European Journal of Operational Research*, Vol. 182, No. 2, PP. 481–501.
4. Wurman, P.R., D’Andrea, R. and Mountz, M. (2008). “Coordinating hundreds of cooperative, autonomous vehicles in warehouses”, *AI Magazine*, Vol. 29, No. 1, PP. 9–20.
5. Guerriero, F., Musmanno, R., Pisacane, O. and Rende, F. (2013). “A mathematical model for the multi-levels product allocation problem in a warehouse with compatibility constraints”, *Applied Mathematical Modeling*, Vol. 37, No. 6, PP. 4385–4398.
6. Muppani, V.R. and Adil G.K. (2006). “Formation of storage classes in the presence of space cost for warehouse planning”, *International Journal of Services Operations and Informatics*, Vol. 1, No. 3, PP. 286–303.
7. Heragu, S.S. (2005). “Mathematical model for warehouse design and product allocation”, *Int. J. Production Research*, Vol. 43, No. 2, PP. 327–338.
8. Önüt, S., Tuzkaya, U.R. and Doga B. (2008). “A particle swarm optimization algorithm for the multiple-level warehouse layout design problem”, *Computers and Industrial Engineering*, Vol. 54, No. 4, PP. 783–799.
9. Sanei, O., Nasiri, V., Marjani, M.R. and Moattar-Husseini, S.M. (2011). “A heuristic algorithm for the warehouse space assignment problem considering operational constraints: with application in a case study”, *Proc. 2011 Int. Conf. on Industrial Engineering and Operations Management*, Kuala Lumpur, Malaysia, January 22–24.
10. Bektas, T. (2006). “The multiple traveling salesman problem: an overview of formulations and solution procedures”, *Omega*, Vol. 34, No. 3, PP. 209–219.
11. Theys, C., Broysy, O., Dullaert, W., Raa. (2010). “Using a TSP heuristic for routing order pickers in warehouses”, *European Journal of Operational Research*, PP. 755–763.
12. Norouzi, N., et al. (2015). “New mathematical modeling for a facilities location and vehicle routing problem solving by a hybrid imperialist competitive algorithm”, Journal of Industrial Engineering, vol 49, No 1 , PP. 129–137.

13. Parragh, S., Doerner, K. and Hart, R. (2008). “A survey on pickup and delivery problems: Part I, Transportation between customers and depot”, *Journal für Betriebswirtschaft*, Vol. 58, No. 1, PP. 21–51.

14. Pandelis, D.G., Karamatsoukis, C. and Kyriakidis, G. (2013). “Single vehicle routing problems with a predefined customer order, unified load and stochastic discrete demands”, *Probability in the Engineering and Informational Sciences*, Vol. 27. No. 1, PP. 1–23.

15. Setak, M., Jalili Bolhassani, S., Karimi, H. and Gorbani, B. (2014). “A node-based mathematical model towards the location routing problem with intermediate replenishment facilities under capacity constraint”, *International Journal of Engineering*, Vol. 27, No. 6, PP. 911–20.

16. Davoodi, M., Abedin, M., Banyassady, B., Khanteimouri, P. and Mohades A. (2013). “An optimal algorithm for two robots path planning problem on the grid”, *Robotics and Autonomous Systems*, Vol. 61, No. 12, PP. 1406–1414.

17. Roozbehani, H. and D’Andrea R. (2011). “Adaptive highways on a grid”, In Pradalier, C., Siegwart, R., and Hirzinger, G. (Eds.), *Robotics Research*, Vol. 70. Springer Berlin/Heidelberg, PP. 661–680.

18. Yu, J. and LaValle, S.M. (2013). “Planning optimal paths for multiple robots on graphs”, *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, May 2013, Kalsruhe, Germany, PP. 3612-3617.

19. Surynek, P. (2010). “An optimization variant of multi-robot path planning is intractable”, *Proc. of the 24thAAAI Conference on Artificial Intelligence*, PP. 1261–1263.

20. Bruno, J., Coffmann, E. and Sethi, R. (1974). “Scheduling independent tasks to reduce mean finishing time”, *Communications of ACM*, Vol. 17, No. 7, PP. 382–387.

21. Parekh, A. (2016). “Kiva Systems–Robotic Warehouse”, Available online at http://hackedgadgets.com/2010/07/10/kiva-systems-robotic-warehouse.

واژه‌های لاتین به ترتیب استفاده در متن

1. AS/RS
2. RFID
3. class-based
4. MRS
5. setup
6. SLAP
7. PAP
8. Vehicle Routing Problem with Backhauls (VRPB)
9. ST-SR-TA