



بکارگیری تکنیک‌های برنامه‌ریزی مسیر به منظور کنترل خودکار ضوابط گردش و توقف خودرو در پارکینگ

اقبال شاکری^۱، حسین تقدس^۲، فائزه امیری پور ابراهیم آباد^۳

۱- عضو هیئت علمی دانشگاه صنعتی امیرکبیر (پلی تکنیک تهران)
eshakeri@aut.ac.ir

۲- عضو هیئت علمی دانشکده مهندسی دانشگاه تهران
htaghaddos@ut.ac.ir

۳- دانشجوی کارشناسی ارشد دانشگاه صنعتی امیرکبیر (پلی تکنیک تهران)
f.amiripour@aut.ac.ir

چکیده

از جمله ضوابط پیچیده و پرچالشی که هم‌اکنون در ایران به صورت دستی کنترل می‌شود ضوابط توقفگاه‌های خودرو و ضوابط کنترل حرکت خودرو در توقفگاه است. نویسندگان مقاله به منظور خودکارسازی کنترل این دسته از ضوابط کوشیده‌اند با استفاده از ظرفیت‌های مدل‌سازی اطلاعات ساختمان (BIM^۱)، اطلاعات موردنیاز را استخراج و با بکارگیری تکنیک‌های برنامه‌ریزی مسیر (PP^۲)، به منظور یافتن مسیر حرکت خودرو، راه‌حلی نوین برای کنترل ضوابط گردش و تردد خودرو در فضای پارکینگ ارائه دهند. در این مسئله نیازی به یافتن بهینه‌ترین مسیر ممکن نیست و وجود حداقل یک مسیر، الزامات ضوابط را تامین می‌کند. مسئله حائز اهمیت امکان‌سنجی مسیر تولیدی است چراکه قیود حرکتی خودرو نیز باید تامین شود؛ برای پاسخگویی به این مسئله، پس از بررسی قابلیت‌های الگوریتم‌های برنامه‌ریزی RRT^۳ در حل این‌گونه مسائل بوسیله تولید درخت با نقاط رندوم، بکارگیری الگوریتم RRT*^{*} به همراه منحنی Reeds and Shepp پیشنهاد گردید.

واژگان کلیدی: کنترل خودکار ضوابط، برنامه‌ریزی مسیر، مدل‌سازی اطلاعات ساختمان (BIM)

^۱ Building Information Modelling

^۲ Path Planning

^۳ Rapidly-exploring Random Tree



۱- مقدمه

عبارت مدل سازی اطلاعات ساختمان هم شامل فرایند ایجاد مدل های دیجیتالی اطلاعات و هم فرایند نگه داری، استفاده و مبادله ی این اطلاعات در طول کل چرخه ی حیات پروژه است. به دلیل توانایی مدل های BIM در پوشش سطوح بالایی از اطلاعات، می توان آن را با تمهیداتی به منظور کنترل مطابقت با ضوابط و آیین نامه های موجود، استفاده کرد. به دلیل پیچیدگی ذاتی، زبانی و محتوایی و ارتباطات درونی فراوان در مقررات و آیین نامه های ساخت، فرایند تطبیق ضوابط که در حال حاضر در کشور ما به صورت دستی انجام می شود، کاری زمان بر، هزینه بر و پر زحمت بوده و بستر مناسبی برای خطاهای انسانی سهوی و عمدی است؛ لذا اهمیت رفع معایب کنترل دستی ضوابط و بهره گیری از محاسن خودکارسازی در این حوضه بر کسی پوشیده نیست. با شناخت بیشتر قابلیت هایی که مدل سازی اطلاعات ساختمان به ارمغان آورده و نیز توسعه بکارگیری آن توسط کاربران، پیچیدگی و جزئیات مدل های تولیدی افزایش یافته است؛ لذا دیگر برای کاربران ممکن نخواهد بود که طبق فرایندهای مرسوم، برای اطمینان از کیفیت و صحت اطلاعات مدل سازی شده، به کنترل دستی و بررسی عینی مدل های BIM بپردازند؛ حال آنکه این مدل ها علاوه بر مشخصات و الزامات معمول، حاوی مفاهیم و اطلاعات انتزاعی و ضمنی نیز هستند و این امر، فرایند کنترل را پیچیده تر خواهد کرد؛ در نتیجه تعریف و پیاده سازی فرایندی برای کنترل خودکار ضوابط، که در آن قوانین تعریف شده بر اساس کدهای ساختمانی، به طور خودکار و با حداقل دخالت انسانی مطابقت مدل ها را کنترل کنند، الزامی خواهد بود. با توجه به تنوع و پیچیدگی ضوابط ساختمانی، الگوریتم کنترل هر ضابطه متفاوت بوده و کنترل خودکار را با چالش های جدیدی روبرو می کند. از جمله ضوابط معماری پیچیده و مهم، ضوابط مربوط به محل های توقف و نیز تردد خودروها در پارکینگ است که به واسطه پیچیدگی و نوع مسئله، استفاده از ظرفیت های علوم جدیدی همچون هوش مصنوعی و رباتیک در کنار الگوریتم های متداول کنترل ضوابط ضروری دانسته شد. در این مقاله ابتدا در بخش ۱-۱ و ۱-۲ کلیات و مفاهیم مورد استفاده بیان می گردد. در بخش ۲ ضوابط یاد شده بررسی و رفع ابهام گردیده و در انتها در بخش ۳ چارچوب پیشنهادی برای کنترل این دسته از ضوابط و نحوه بهره گیری از قابلیت های BIM و PP برای حل مسئله ارائه می گردد.

۱-۱ فرایند کنترل خودکار ضوابط

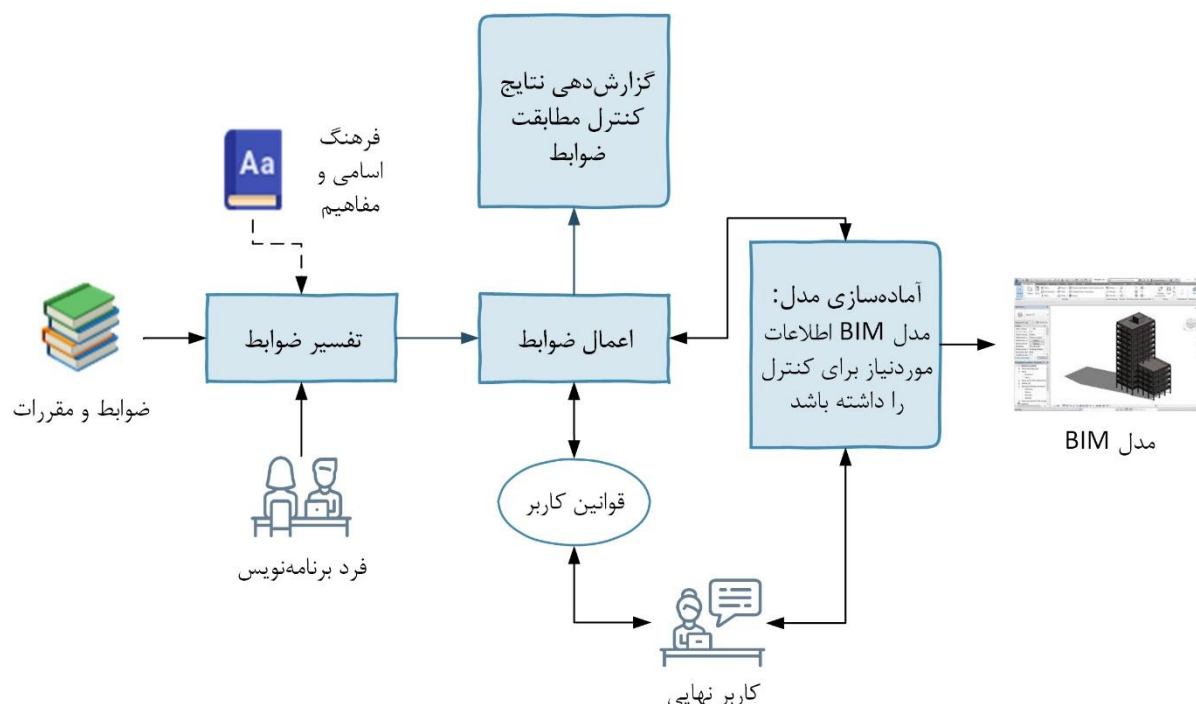
تفسیر ضوابط ساختمانی به نحوی که برای کامپیوتر قابل فهم باشد و بتواند روی آن عملیات انجام دهد، سابقه ای طولانی دارد. تا به این لحظه بسترهای نرم افزاری متنوعی برای اعمال کنترل خودکار ضوابط در کشورهای مختلف پیاده سازی شده است. این بسترها سطوح متفاوتی را در زمینه ی توانایی خودکار کردن کنترل های فاز طراحی فراهم کرده و می توانند به صورت نرم افزاری مستقل، افزونه ای برای یکی از نرم افزارهای طراحی و یا برنامه ای تحت وب باشند.

Eastman و همکاران در سال ۲۰۰۹، ساختاری ۴ مرحله ای شامل (۱) تفسیر قوانین و ساختار منطقی آنها، (۲) آماده سازی مدل، (۳) اعمال ضوابط و (۴) گزارش گیری، به منظور کنترل خودکار ضوابط معرفی نمودند (شکل ۱) (Nawari, 2018). با وجود امکانات گسترده ای که BIM فراهم آورده، همچنان موانعی بر سر راه کنترل تمام خودکار ضوابط وجود دارد. در راستای استفاده از مدل اطلاعات ساختمان در فرایند کنترل خودکار ضوابط دو رویکرد ¹ API نویسی و استفاده از مدل داده ² IFC قابل استفاده است. در رویکرد اول، با استفاده از API یک نرم افزار BIM، مجموعه ای از توابع آن نرم افزار مورد استفاده

¹ Application Programming Interface

² Industry Foundation Classes

قرار می گیرد و قابلیت های جدیدی به آن اضافه می شود؛ از جمله توسعه سیستم کنترل خودکار ضوابطی که تنها محدود به همان نرم افزار خواهد بود. در رویکرد IFC باید از مدل BIM ساخته شده در هر یک از نرم افزارهای BIM خروجی IFC گرفته شود و سپس اطلاعات مورد نیاز از قالب IFC استخراج گردد (شاکری و همکاران، ۱۳۹۸). تقریباً تمامی سیستم های کنترل خودکار موفق در جهان از رویکرد دوم استفاده کرده (Eastman *et al.*, 2009) و بدین سبب و نیز به دلیل عدم اجبار به استفاده از نرم افزار مدل سازی خاص، در این پژوهش نیز از رویکرد IFC استفاده می شود.



شکل (۱) کنترل خودکار ضوابط در بستر مدل سازی اطلاعات ساختمان.

۱-۲ برنامه ریزی مسیر و رویکردهای آن

برنامه ریزی مسیر شاخه مهمی از پژوهش های مربوط به علم رباتیک و سیستم های بدون سرنشین است. مسئله مسیریابی بهینه از نیاز به بهینه سازی مواردی از جمله هزینه، طول مسیر جابجایی، زمان سفر و ... نشئت گرفته است، تا با استفاده از آن بتوان مسیر بهینه ای را جستجو کرد که متحرک را بدون برخورد با موانع از نقطه مبدا به مقصد پیش بینی شده هدایت کرد. در یک تقسیم بندی مشابه آن چه در شکل ۲ نشان داده شده است، الگوریتم های برنامه ریزی مسیر در سه دسته ی (۱) الگوریتم های ترکیبی، (۲) الگوریتم های نمونه محور و (۳) الگوریتم های الهام گرفته از طبیعت قرار می گیرند. برخلاف الگوریتم های نمونه محور که بر اساس احتمالات عمل می کنند، الگوریتم های ترکیبی پاسخی دقیق ارائه می کنند (Debnath *et al.*, 2019). الگوریتم های ترکیبی اغلب شامل دو سازوکار تولید فضای پیکربندی^۱ و بکارگیری الگوریتم های جستجو در گراف هستند. نمایش

¹ Configuration space method



چهارمین کنفرانس بین المللی مدل سازی اطلاعات ساخت

فضای پیکربندی تولیدشده به کمک تکنیک‌هایی همچون Road map، تجزیه به سلول‌ها^۱ و میدان پتانسیل^۲ انجام می‌شود (Debnath *et al.*, 2019). زمانی که پیکربندی‌های فضایی شروع و هدف حرکت مشخص گردید، الگوریتم‌های جستجو در گراف بکار گرفته شده تا بهینه‌ترین مسیر از نقطه شروع به نقطه هدف محاسبه و انتخاب گردد.

اول بار Lozano-Perez در سال ۱۹۸۳ فضای پیکربندی را به‌عنوان راه‌حل مسئله یافتن مسیر عبور جسم متحرک محدب دو بعدی، از میان موانع محدب دو بعدی مطرح کرد. هدف اصلی کاهش ابعاد جسم متحرک از یک چندضلعی به یک نقطه به‌واسطه اصلاح موانع و تبدیل مسئله مسیریابی شی دو بعدی، به مسئله حرکت یک نقطه از میان موانع دو بعدی است.

ایده اصلی الگوریتم‌های نمونه‌محور، اتصال تعدادی نمونه رندوم در فضای خارج از موانع است و تفاوت روش‌های متفاوت آن از جمله PRM و RRT در نحوه اتصال این نقاط نمونه است. این الگوریتم‌ها خصوصاً در فضاهای پیکربندی با درجات بالای آزادی عملکرد مطلوبی داشته‌اند (Kang *et al.*, 2016)؛ اما این مسئله که در صورت وجود مسیر این الگوریتم‌ها قادر به یافتن حداقل یک مسیر هستند، اما این مسیر لزوماً بهینه‌ترین مسیر نیست، می‌تواند عملکرد آن‌ها را محدود کند (Elbanhawi and Simic, 2014). از جمله الگوریتم‌های برنامه‌ریزی مسیر الهام‌گرفته از طبیعت می‌توان الگوریتم ژنتیک^۳، بهینه‌سازی ازدحام ذرات^۴ و بهینه‌سازی کلونی مورچه‌ها^۵ را نام برد. GA برای مسیریابی از فرایند انتخاب طبیعی تقلید می‌کند. در این میان الگوریتم PSO به‌منظور حل مسئله مسیریابی در مدل‌های چندهدفه عملکرد خوبی داشته‌است (Davoodi *et al.*, 2015). در الگوریتم ACO با تقلید از مسیریابی مورچه‌ها در حالتی که محل غذا مشخص است و مورچه‌ها با ترشح آنزیم فرومون^۶ مسیر رسیدن به غذا را نشانه‌گذاری می‌کنند، به مسیریابی می‌پردازد و به‌طور تصادفی مسیری بین شروع و هدف پیدا می‌کند (Wang *et al.*, 2011).

مدلی که اطلاعات دقیق معنایی، هندسی و ارتباطات مکانی موردنیاز برای برنامه‌ریزی مسیر را داشته باشد می‌تواند از نوع مدل BIM در فرمت IFC و یا مدل GIS در فرمت‌های CityGML و IndoorGML باشد. در سطح اول IFC و پس از آن CityGML اطلاعات هندسی و معنایی غنی‌ای را برای مسیریابی درون ساختمان فراهم می‌کنند؛ گرچه از منظر ضمنی نبودن اطلاعات توپولوژیک، IndoorGML مناسب‌تر است اما استفاده از آن چندان متداول نیست و اطلاعات محدودی دارد؛ لذا به دلیل بلوغ و جامعیت مدل IFC، استفاده از آن رواج بیشتری دارد (Hamieh *et al.*, 2020). Liu و همکاران در سال ۲۰۲۱، چالش‌های استفاده از IFC برای مسیریابی و امکان پیاده‌سازی روش‌هایی همچون MAT و VG برای تولید گراف را بررسی کردند (Liu *et al.*, 2021). در پژوهش دیگری، Lin و Lin تنها به اطلاعات هندسی بدست‌آمده از طرح IFC اکتفا کرده و پس از تفکیک فضاها از روش مثلث‌سازی Delaunay برای ساخت گراف استفاده کردند (Lin and Lin, 2018). دسته دیگر پژوهش‌های این حوزه، تنها منحصر به محیط ساختمان نبوده تا امکان مطالعه طیف وسیع‌تری از انواع مسائل برنامه‌ریزی و انتخاب راه‌حلی مناسب را فراهم کند؛ از جمله مقالاتی که به برنامه‌ریزی مسیر ربات‌های شبه‌ماشین اختصاص دارند (Candeloro and Lekkas, 2013; Masehian and Kakahaji, 2014; Yang *et al.*, 2014).

¹ Cell decomposition techniques

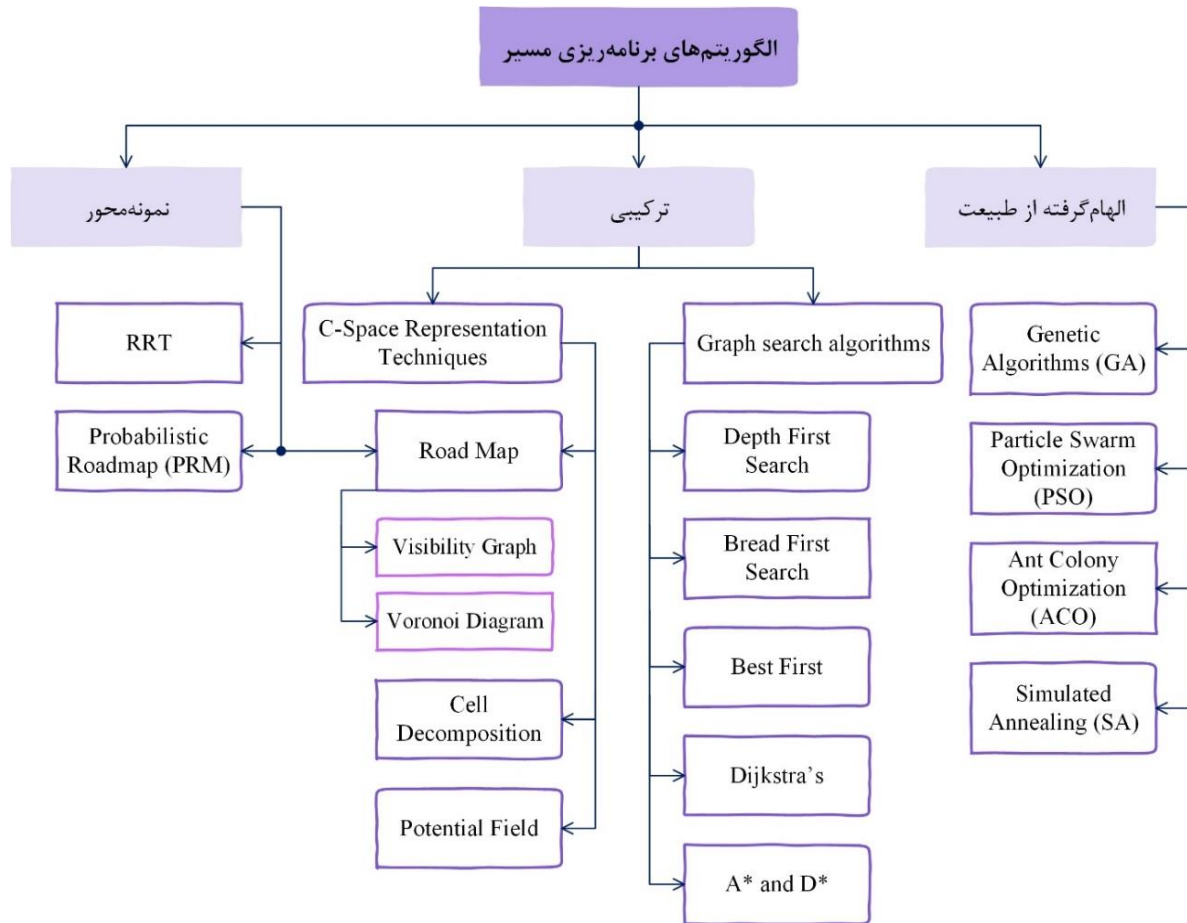
² Potential field

³ Genetic Algorithm (GA)

⁴ Particle Swarm Optimization (PSO)

⁵ Ant Colony Optimization (ACO)

⁶ Pheromones



شکل (۲) تقسیم‌بندی الگوریتم‌های برنامه‌ریزی مسیر.

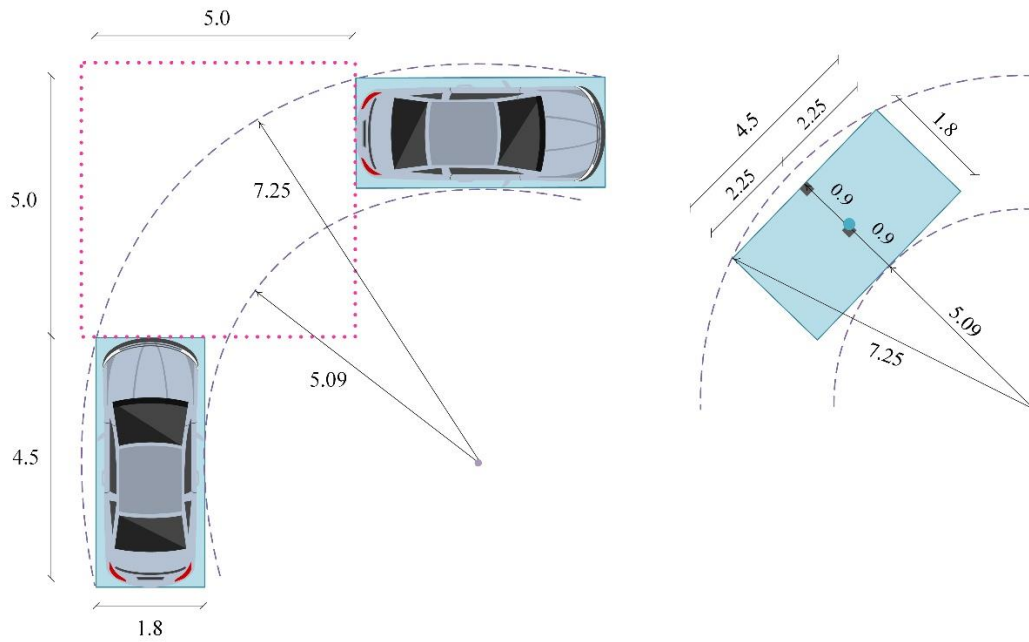
۲- ضوابط حرکت و گردش خودرو در پارکینگ

بخش‌هایی از مبحث چهارم مقررات ملی ساختمان و طرح تفصیلی به ضوابط توقفگاه‌های خودرو اختصاص دارد. بندهای ۴-۱۰-۲-۲ و ۴-۱۰-۵-۲ از مبحث چهارم مقررات ملی ساختمان که به ترتیب به موضوعات شعاع درونی مسیر چرخش خودرو و رعایت گردش ۹۰ درجه خودرو اختصاص دارند، برای خودکارسازی در این مقاله در نظر گرفته شده‌اند؛ بعلاوه کنترل بند ۴-۱۰-۵-۲ نیز که به موضوع ابعاد و مساحت محل‌های توقف خودرو اختصاص دارد، به واسطه ارتباطی که با ضوابط تردد دارد، ضروری است.

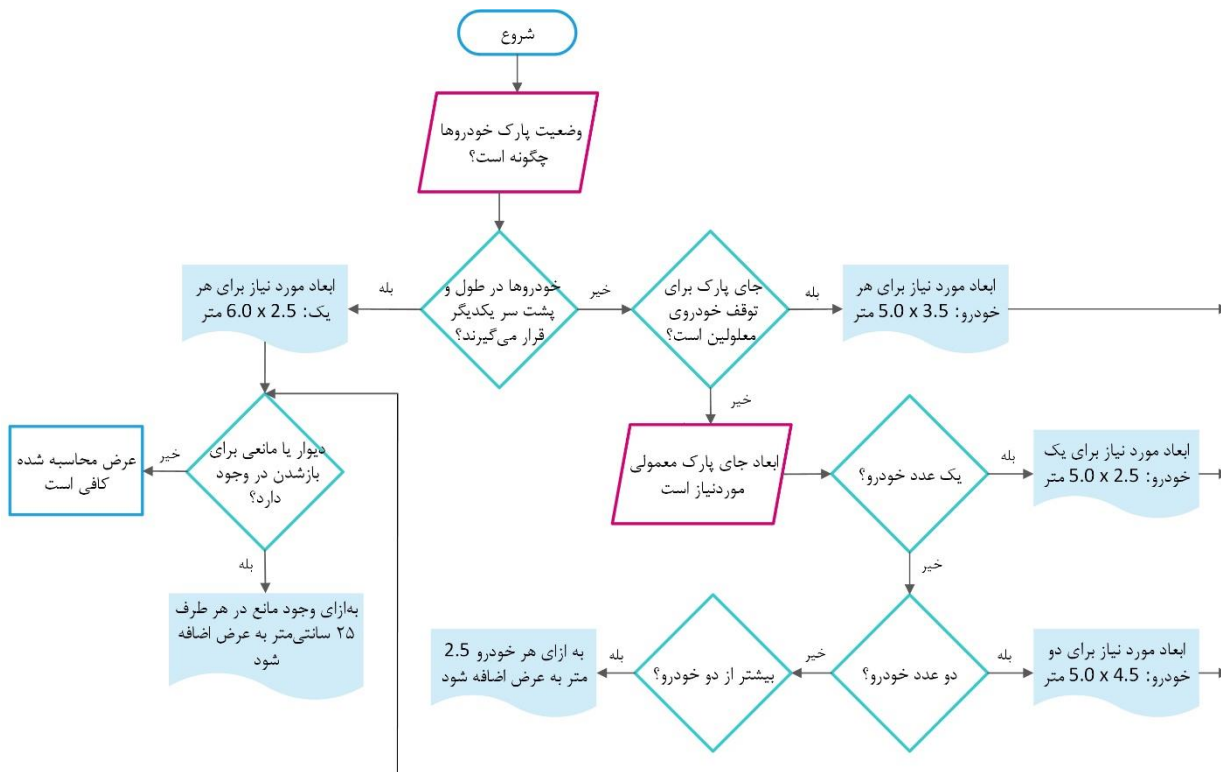
در بند ۴-۱۰-۲-۲ شعاع درونی مسیر چرخش خودرو، حداقل ۵ متر ذکر شده و در بند ۴-۱۰-۵-۲ فضای با حداقل ابعاد ۵*۵ متر برای گردش خودرو الزامی دانسته شده‌است؛ همانطور که در شکل ۳ نمایش داده شده با فرض ابعاد ۱.۸*۴.۵ متر برای خودرو، شعاع گردش مرکز خودرو قابل محاسبه خواهد بود. در بند ۴-۱۰-۵-۲ ابعاد لازم جهت توقف یک یا چند خودرو در حالت مجاور، پشت سرهم و همچنین برای خودرو معلول ذکر شده و در فلوچارت شکل ۴ نمایش داده شده‌است.



چهارمین کنفرانس بین المللی مدل سازی اطلاعات ساخت



شکل (۳) حداقل ابعاد فضای گردش ۵*۵ و محاسبه شعاع گردش خودرو بر اساس آن.



شکل (۴) فلوچارت تعیین ابعاد لازم جهت توقف خودرو طبق بند ۴-۵-۱۰-۲-۴.



۳- کنترل خودکار ضوابط حرکت خودرو در پارکینگ

کنترل خودکار ضوابط مربوط به گردش خودرو به واسطه پیچیدگی‌هایی که دارد در قالب کنترل‌های ساده هندسی نمی‌گنجد و از آنجا که مسئله یافتن مسیر حرکت یک جسم متحرک از یک نقطه شروع و عبور از میان موانع و رسیدن به نقطه پایان مشخص، یعنی دقیقاً آنچه مقصود این پژوهش بوده، همان تعریف برنامه‌ریزی مسیر در علم رباتیک است، نویسندگان مقاله بر آن شدند که از الگوریتم‌های مسیر استفاده کرده و صرف وجود یا عدم وجود مسیر را از نقطه شروع حرکت تا موقعیت پارک خودرو بررسی کنند؛ چراکه وجود چنین مسیری اثباتی بر امکان گذر خودرو از مسیرهای عبور و وجود فضای کافی برای گردش خواهد بود (شکل ۵).



شکل (۵) چارچوب پیشنهادی کنترل خودکار ضوابط مسیر حرکت خودرو در پارکینگ

۳-۱ اصلاح مدل اطلاعات ساختمان متناسب با نیاز ضوابط

در بخش‌های ۱-۱ و ۲-۱ با استناد به مقالاتی، استفاده از مدل‌های BIM و دریافت خروجی آن‌ها در فرمت IFC به‌عنوان راهی برای دسترسی به اطلاعات مورد نیاز کنترل ضوابط پارکینگ مناسب دانسته شد. به‌منظور کنترل این دسته از ضوابط، وجود



نمایشی از محیط داخلی ساختمان شامل اطلاعات هندسی فضای پارکینگ، موانع (دیوار، ستون، سایر کامپوننت‌ها و...)، درگاه‌ها (رمپ ورود به پارکینگ یا درب ورود خودرو) و نیز حاوی ارتباطات مکانی و معنایی کافی باشد، الزامی است. مدل سازی اطلاعات ساختمان به دلیل پوشش سطوح بالایی از اطلاعات، اطلاعات مورد نیاز را فراهم می‌کند. لازم به ذکر است که بایستی پیش از دریافت خروجی، در مرحله آماده‌سازی مدل ساختمان، کامپوننت‌هایی با ابعاد استاندارد به مدل‌ها اضافه گردد تا در کنترل ابعاد از آن‌ها استفاده شود.

۲-۳ استخراج اطلاعات از طرح داده IFC

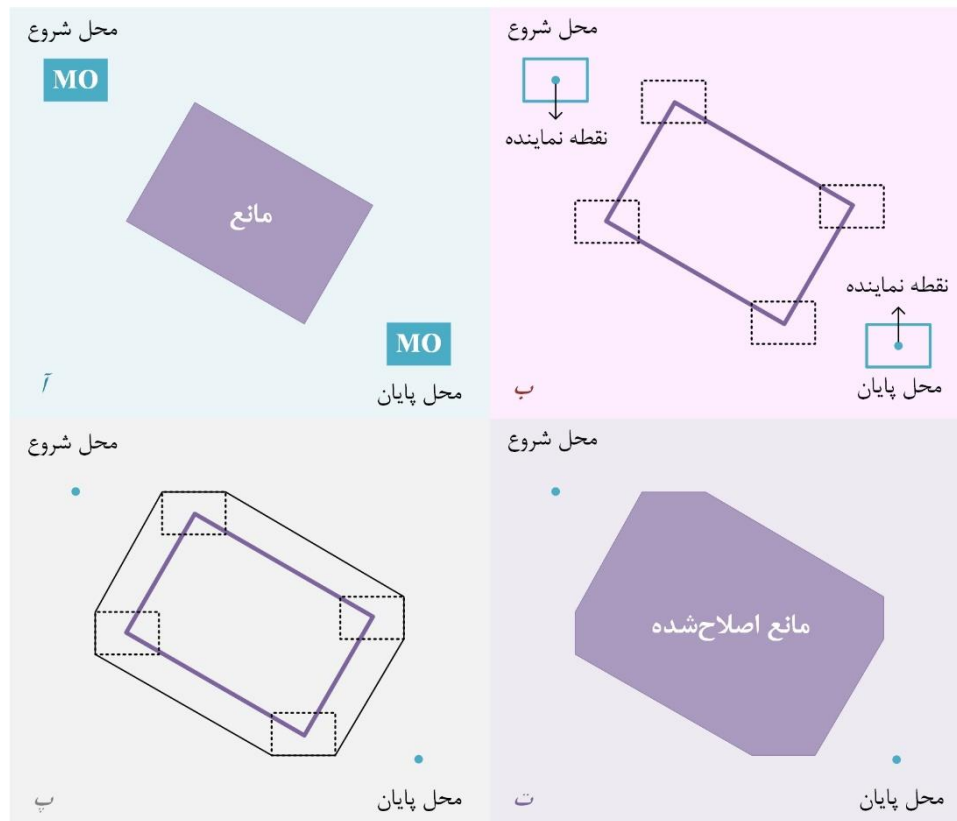
با دسترسی به خروجی IFC مدل ساختمان، اطلاعات مورد نیاز شامل فضای طبقات پارکینگ، موانع شامل دیوارها، ستون‌ها، کامپوننت‌های پارک و ... استخراج می‌گردد؛ بدین منظور کدهای برنامه‌نویسی به زبان پایتون و در محیط برنامه‌نویسی Jupyter notebook، با استفاده از کتابخانه IfcOpenShell نوشته می‌شود.

۳-۳ انتخاب الگوریتم و پیاده‌سازی

• ساخت فضای پیکربندی

فرض شده حرکت خودرو در پارکینگ دارای سه درجه آزادی باشد، دو درجه آزادی مکانی در صفحه در راستای محورهای X, Y و یک درجه جهت‌گیری ماشین نسبت به یکی از محورها؛ از آنجا که خودروها، وسایل نقلیه غیرهولونومیک^۱ (سیستمی با قید سرعتی انتگرال‌ناپذیر) هستند، این قیود نیز باید در فضای پیکربندی اعمال گردد (Paromtchik and Laugier, 1996). در گام نخست به منظور تولید فضاهای پیکربندی باید ابعاد خودرو استاندارد و نقطه نماینده، مشخص شود (شکل ۳). سپس وضعیت‌های شروع (بسته به جهت پارک خودرو، مختصات نقطه نماینده و جهت آن) و نیز وضعیت پایان مسیر (درب ورودی خودرو یا رمپ ورودی طبقه) مشخص می‌گردد. هر بار یکی از کامپوننت‌ها برای مسیریابی انتخاب و دیگر کامپوننت‌ها مانع در نظر گرفته می‌شود. به منظور تولید فضای پیکربندی برای هر جهت‌گیری خودرو، مشابه زیر عمل می‌شود. در مثال نمایش داده شده در شکل ۶، هدف یافتن مسیر بدون برخورد با مانع برای حرکت جسم از محل شروع و رسیدن به محل پایان است. طبق توضیحاتی که در مورد روش فضای پیکربندی داده شد، نقطه‌ای به نمایندگی از جسم متحرک انتخاب می‌شود؛ سپس پولیگان جسم متحرک را از محل نقطه نماینده روی محیط مانع حرکت داده و شکل محدب حاصل، شکل اصلاح شده مانع با توجه به جهت‌گیری متحرک، مانع پیکربندی یا C-Obstacle نام دارد. به محیط حاصل یک فضای پیکربندی یا C-Space گفته می‌شود؛ فضایی که تمامی چیدمان‌های ممکن را نمایش دهد تا بتوان در هر لحظه مکانی عاری از مانع را برای نقطه متحرک برگزید (Kayhani et al., 2021).

¹ Nonholonomic



شکل (۶) اصلاح شکل موانع در فرایند تولید فضای پیکربندی برای یک جهت گیری خاص جسم متحرک (Kayhani et al., 2021).

• انتخاب الگوریتم برنامه ریز با در نظر گرفتن قیود سرعتی حرکت ماشین

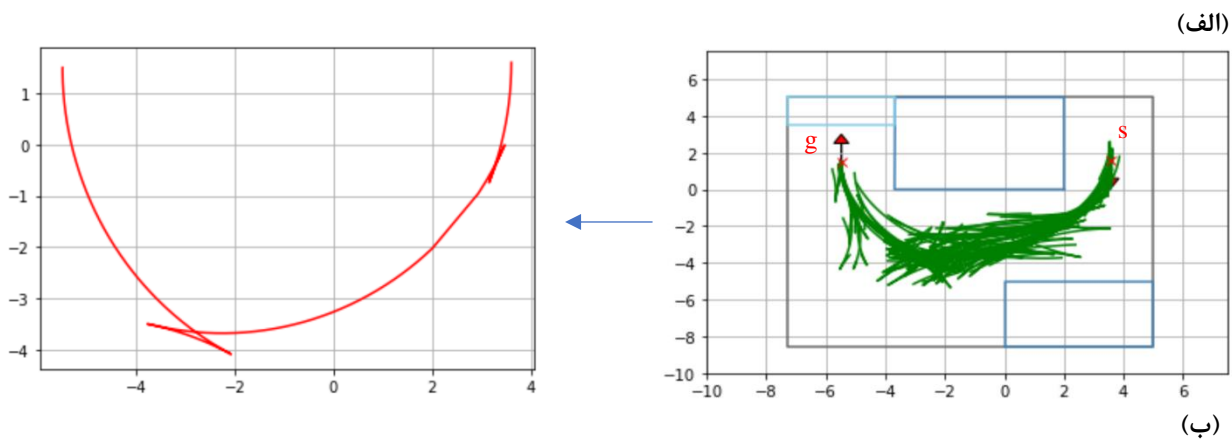
درک محدودیت‌های مانوردهی ربات‌ها به دلیل اطمینان از امکان‌سنجی حرکت در مسیر تولیدی مهم است (Yang et al., 2014). الگوریتم‌های RRT به طور خاص برای پشتیبانی از برنامه ریزی مسیر حرکت وسایل نقلیه غیرهولونومیک در درجات بالای آزادی طراحی شده‌اند؛ در واقع این الگوریتم‌ها به دلیل اینکه نیازی به از پیش مشخص شدن هیچ یک از ارتباطات میان پیکربندی‌ها ندارند، می‌توان آن‌ها را مستقیماً در این گونه مسائل به کار بست (Lavalle, 1998).

فرایند مسیریابی توسط الگوریتم RRT بدین صورت است که در فضای جستجو (Z) با نمونه‌های رندوم درختی ایجاد می‌شود که ریشه در نقطه شروع ($Z_{init} \in Z$) دارد و به تدریج با تکرارهای بیشتر رشد کرده تا به نقطه هدف (Z_{goal}) برسد. در هر تکرار یک نقطه رندوم (Z_{rand}) از فضای آزاد پیکربندی (Z_{free} ، که شامل موانع نمی‌باشد) انتخاب می‌شود؛ سپس الگوریتم به دنبال نزدیک‌ترین نقطه ($Z_{nearest}$) از درخت به نقطه رندوم می‌گردد که این نزدیکی بسته به فرضیات مسئله تعریف می‌شود. اگر Z_{rand} با گام از پیش تعریف شده از طریق $Z_{nearest}$ قابل دستیابی باشد، در درخت این دو نقطه با شاخه‌ای به هم متصل می‌گردند؛ در

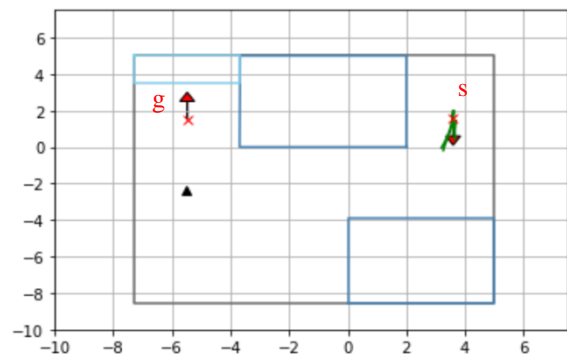


چهارمین کنفرانس بین المللی مدل سازی اطلاعات ساخت

غیر این صورت با استفاده از تابع فرمان^۱ یک نقطه جدید (Z_{new}) بر می گرداند و درخت برای اتصال $Z_{nearest}$ با این نقطه رشد داده می شود. هر بار در پروسه شاخه سازی درخت به سوی نقطه جدید، عدم برخورد آن با موانع توسط یک تابع بولی کنترل می شود. تا زمان رسیدن به تعداد تکرار از پیش تعریف شده این مراحل تکرار می گردند (Noreen, Khan and Habib, 2016). الگوریتم RRT^* یکی از انواع RRT ها می باشد که به واسطه افزودن قابلیت هایی جدید به RRT ، بهبود چشم گیری در کیفیت مسیر تولیدی پیدا کرده است؛ به این نحو که در صورت معتبر بودن نقطه جدید، این نقطه مستقیماً به درخت افزوده نمی شود بلکه کره ای با شعاع معین در اطراف این نقطه تعریف شده و برای نقاطی از درخت که هزینه آن ها بیشتر از هزینه اتصال آن ها با نقطه جدید است، نقطه والد به نقطه جدید تغییر یافته و مسیرشان اصلاح می گردد (Karaman and Frazzoli, 2007). به منظور اعمال قیود حرکتی خودرو، تابع فرمان مورد استفاده در الگوریتم RRT^* تابع Reeds and Shepp (R&S) می باشد. Fraichard و Scheuer این روش را برای پیش بینی مختصات نقاط پیموده شده توسط ماشینی که قابلیت حرکت رو به جلو و عقب داشته باشد و بر اساس عدم تجاوز از حداکثر انحنای مجاز چرخش کند، معرفی نموده اند (Fraichard et al., 2005). در شکل های ۷-الف و ۷-ب محیط های نمونه نمایش داده شده است. در این محیط ها فضای کلی و موانع با پولیگان های به رنگ های طوسی و آبی و وضعیت در شروع و پایان نیز با S و g نشان داده شده است. تابع فرمان R&S و تابع هزینه، مسافت طی شده بوده است. منحنی های سبز رنگ، شاخه های درخت تولیدی الگوریتم RRT و منحنی قرمز مسیر نهایی است. در شکل ۷-الف ابعاد لازم برای چرخش (فضای 5×5 متر) تامین شده و تولید مسیر نهایی اثباتی بر این ادعاست ولی اگر فضای کافی برای چرخش موجود نباشد (شکل ۷-ب)، مسیری تولید نشده و ضابطه نقض می شود. هر نقطه از مسیر، نمایشگر نقطه وسط چرخ های عقب خودرو بوده و در کنترل های عدم برخورد، با توجه به جهت گیری خودرو، عدم برخورد این نقطه با موانع در فضای پیکربندی مربوطه کنترل می گردد.



¹ Steering Function



شکل (۷) محیط های طراحی شده نمونه برای ارزیابی عملکرد الگوریتم RRT و منحنی R&S.

۴- نتیجه گیری

فرایند کنترل دستی ضوابط امری زمان بر، هزینه بر و مستعد خطاهای انسانی بوده و خودکارسازی آن موجب افزایش بهره‌وری و کاهش فساد اداری خواهد بود. در این مقاله در راستای خودکارسازی ضوابط از تکنیک‌هایی همچون مدل‌سازی اطلاعات ساختمان و برنامه‌ریزی مسیر به منظور پیشنهاد راهکاری برای کنترل خودکار ضوابط گردش و حرکت خودرو در پارکینگ استفاده گردید و بیان شد که اگر بتوان با الگوریتم برنامه‌ریز مسیری، حداقل یک مسیر برای حرکت خودرو از محل شروع تا توقف در محل پارک یافت، اثباتی بر تامین ضوابط مذکور است. این مسیر باید ویژگی‌های داشته باشد که در بخش ۳ به تفصیل توضیح داده شد.

ملاحظات این مقاله در کنترل خودکار ضوابط حرکت خودرو در پارکینگ به اختصار عبارتند از:

- اطلاعات هندسی و مکانی مورد نیاز از خروجی فرمت IFC مدل ساختمان تامین می‌گردد.
- ابعاد خودرو و شعاع گردش آن باید با ابعاد مجاز ضوابط برای گردش و ... سازگار باشد.
- خودرو وسیله نقلیه غیرهولونومیک است و برای تامین قیود حرکتی آن باید از ابتدا این قیود را دخیل کرد یا با روش‌هایی مسیر تولید شده را متناسب با آن‌ها اصلاح نمود.
- تولید فضای پیکربندی سبب تسهیل مسیریابی می‌گردد؛ اما بدلیل غیرهولونومیک بودن حرکت خودرو، تعیین ارتباطات بین پیکربندی‌ها دشوار است که با استفاده از الگوریتم‌های RRT این مسئله رفع می‌شود.
- به منظور تامین قیود حرکتی خودرو، از منحنی R&S به عنوان تابع فرمان در الگوریتم RRT* استفاده گردید.

به این دلیل که کنترل ضوابط حتی الامکان باید سریع صورت بگیرد، در ادامه این مقاله، پژوهش‌هایی در زمینه تعیین تعداد تکرار بهینه در الگوریتم RRT*، استفاده از انواع دیگر الگوریتم RRT از جمله Bidirectional RRT که همزمان از نقطه شروع و پایان، دو درخت را رشد داده و به محض برخورد درخت‌ها، مسیر تولید شده ارائه می‌گردد، توصیه می‌شود. به علاوه در این مقاله از محاسن استفاده از مدل‌سازی اطلاعات ساختمان و نحوه استخراج اطلاعات مورد نیاز از آن برای برنامه‌ریزی مسیر سخن گفته شد؛ سپس در ادامه در محیط‌های نمونه کارکرد الگوریتم انتخابی سنجیده شد؛ لذا لازم است این الگوریتم‌ها در مدل‌های BIM به شرح گفته شده پیاده‌سازی و اعتبارسنجی گردند.

تشکر و قدردانی



نویسندگان این مقاله مراتب تشکر و قدردانی خود را از معاونت شهرسازی و معماری شهرداری تهران و مرکز تحقیق و توسعه تکنولوژی‌های نوین ساخت (تکنوسا) به جهت همکاری صمیمانه در جمع‌آوری اطلاعات اعلام می‌دارند.

منابع و مراجع

- [1] شاکری، ا.، تقدس، ح.، بابایی راوندی، ا.، عباسیان‌فر، و. (۱۳۹۸) "ارائه چارچوبی به منظور کنترل خودکار ضوابط ساخت و ساز در بستر مدل سازی اطلاعات ساختمان (BIM)", دومین کنفرانس ملی مدل سازی اطلاعات ساختمان.
- [2] مبحث چهارم مقررات ملی ساختمان (۱۳۹۶) "الزامات عمومی ساختمان", دفتر تدوین و ترویج مقررات ملی ساختمان
- [3] Borrmann, A. *et al.* (2018) *Building Information Modeling, Technology Foundations and Industry Practice*. Springer.
- [4] Candeloro, M. and Lekkas, A. M. (2013) 'Continuous Curvature Path Planning using Voronoi diagrams and Fermat's spirals', (1).
- [5] Davoodi, M. *et al.* (2015) 'Clear and smooth path planning', *Applied Soft Computing Journal*. Elsevier B.V., pp. 1–12. doi: 10.1016/j.asoc.2015.04.017.
- [6] Debnath, S. K. *et al.* (2019) 'Comparison of different configuration space representations for path planning under combinatorial method', *Indonesian Journal of Electrical Engineering and Computer Science*, 14(1), pp. 1–8. doi: 10.11591/ijeecs.v14.i1.pp1-8.
- [7] Eastman, C. *et al.* (2009) 'Automatic rule-based checking of building designs', *Automation in Construction*. Elsevier B.V., 18(8), pp. 1011–1033. doi: 10.1016/j.autcon.2009.07.002.
- [8] Elbanhawi, S. and Simic, M. (2014) 'Sampling-Based Robot Motion Planning: A Review', *IEEE Access*. IEEE, 2, pp. 56–77. doi: 10.1109/ACCESS.2014.2302442.
- [9] Fraichard, T. *et al.* (2005) 'From Reeds and Shepp's to continuous-curvature paths', *IEEE Transactions on Robotics*. IEEE, 20(6), pp. 1025–1035.
- [10] Hamieh, A. *et al.* (2020) 'A BIM-based method to plan indoor paths', *Automation in Construction*. Elsevier, 113(July 2019), p. 103120. doi: 10.1016/j.autcon.2020.103120.
- [11] Kang, G. *et al.* (2016) 'Sampling-based Path Planning with Goal Oriented Sampling', in *IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*. IEEE, pp. 1285–1290. doi: 10.1109/AIM.2016.7576947.
- [12] Karaman, S. and Frazzoli, E. (2007) 'Sampling-based Algorithms for Optimal Motion Planning', pp. 1–76.
- [13] Kayhani, N. *et al.* (2021) 'Heavy mobile crane lift path planning in congested modular industrial plants using a robotics approach', *Automation in Construction*. Elsevier B.V., 122(December 2020), p. 103508. doi: 10.1016/j.autcon.2020.103508.
- [14] Lavalle, S. M. (2006) *Planning Algorithms*. Cambridge University Press.
- [15] Lin, W. Y. and Lin, P. H. (2018) 'Intelligent generation of indoor topology (i-GIT) for human indoor path finding based on IFC models and 3D GIS technology', *Automation in Construction*. Elsevier, 94(December 2017), pp. 340–359. doi: 10.1016/j.autcon.2018.07.016.
- [16] Liu, L. *et al.* (2021) 'Indoor navigation supported by the Industry Foundation Classes (IFC): A survey', *Automation in Construction*. Elsevier B.V., 121(September 2020), p. 103436. doi: 10.1016/j.autcon.2020.103436.
- [17] Masehian, E. and Kakahaji, H. (2014) 'NRR : a nonholonomic random replanner for navigation of car-like robots in unknown environments', *Robotica*. Cambridge University Press. doi: 10.1017/S0263574713001276.



- [18] Nawari, O. N. (2018) *Building Information Modeling, Automated Code Checking and Compliance Processes*. CRC Press.
- [19] Noreen, I., Khan, A. and Habib, Z. (2016) 'A Comparison of RRT , RRT * and RRT * -Smart Path Planning Algorithms', *IJCSNS International Journal of Computer Science and Network Security*, 16(10), pp. 20–27.
- [20] Paromtchik, I. E. and Laugier, C. (1996) 'Autonomous Parallel Parking of a Nonholonomic Vehicle', in *Proceedings of Conference on Intelligent Vehicles*. IEEE, pp. 13–18.
- [21] Wang, X. *et al.* (2011) 'Collision-Free Path Planning For Mobile Cranes Based On Ant Colony Algorithm', *Key Engineering Materials*, 469, pp. 1108–1115. doi: 10.4028/www.scientific.net/KEM.467-469.1108.
- [22] Yang, K. *et al.* (2014) 'Spline-Based RRT Path Planner for Non-Holonomic Robots', *Intell Robot Syst.* Springer, pp. 763–782. doi: 10.1007/s10846-013-9963-y.

Using Path Planning Techniques for Automatic Code Compliance Checking

Abstract

Controlling parking area and car maneuver dimensions is one of the most complicated and challenging rules in Iran's building regulations which is done manually currently. To automatically control such regulations, articles's authors have made an attempt to extract required information by means of building information modelling (BIM) and have presented a new solution to control the turning and crossing rules in parking areas, using path planning techniques in order to find car's path. There is no need to find the optimal path in this case and existence of at least one path meets the requirements. Since the car's motion constraints must be provided, feasibility of this path is noteworthy. To provide a solution, the capabilities of RRT algorithms were studied and using the RRT* algorithm along with Reeds and Shepp as the steering function was suggested.

Keywords: Automatic Code Compliance Checking, Path Planning, Building Information Modelling (BIM)