بسم الله الرحمن الرحیم

دانشگاه تربیت مدرّس
دانشکده مهندسی برق و کامپیوتر

رساله دکتری در رشته مهندسی کامپیوتر
گرایش نرم‌افزار

یادگیری تقویتی عمیق با شکل‌دهی تصادفی پاداش برای مدیریت عدم قطعیت در سیستم‌های خودوفقی

علیرضا کاویانی‌فر

استاد راهنما

دکتر ...

استاد مشاور

دکتر ...

بهمن ۱۴۰۴

تأییدیه‌ی هیأت داوران جلسه‌ی دفاع از رساله

خانم/آقای علیرضا کاویانی‌فر رساله — واحدی خود را با عنوان یادگیری تقویتی عمیق با شکل‌دهی تصادفی پاداش برای مدیریت عدم قطعیت در سیستم‌های خودوفقی در تاریخ بهمن ۱۴۰۴ ارائه کردند. اعضای هیات داوران نسخه نهایی این رساله را از نظر فرم و محتوا تایید کرده است و پذیرش آنرا برای تکمیل درجه دکتری پیشنهاد می‌کنند.

آیین‌نامه حق مالکیت مادی و معنوی در مورد نتایج پژوهشهای علمی‌دانشگاه تربیت مدرس

مقدمه: با عنایت به سیاست‌های پژوهشی و فناوری دانشگاه در راستای تحقق عدالت و کرامت انسانها که لازمه شکوفایی علمی‌و فنی است و رعایت حقوق مادی و معنوی دانشگاه و پژوهشگران، لازم است اعضای هیأت علمی، دانشجویان، دانش‌آموختگان و دیگر همکاران طرح، در مورد نتایج پژوهشهای علمی‌که تحت عناوین پایان‌نامه، رساله و طرحهای تحقیقاتی با هماهنگی دانشگاه انجام شده است، موارد زیر را رعایت نمایند:

ماده ۱ - حق نشر و تکثیر پایان‌نامه/رساله و درآمدهای حاصل از آنها متعلق به دانشگاه می‌باشد ولی حقوق معنوی پدید آورندگان محفوظ خواهد بود.

ماده ۲ - انتشار مقاله یا مقالات مستخرج از پایان‌نامه/ رساله به صورت چاپ در نشریات علمی‌و یا ارائه در مجامع علمی باید به نام دانشگاه بوده و با تایید استاد راهنمای اصلی، یکی از اساتید راهنما، مشاور و یا دانشجو مسئول مکاتبات مقاله باشد. ولی مسئولیت علمی‌مقاله مستخرج از پایان نامه و رساله به عهده اساتید راهنما و دانشجو می‌باشد. تبصره: در مقالاتی که پس از دانش‌آموختگی بصورت ترکیبی از اطلاعات جدید و نتایج حاصل از پایان‌نامه/ رساله نیز منتشر می‌شود نیز باید نام دانشگاه درج شود.

ماده ۳- انتشار کتاب، نرم‌افزار و یا آثار ویژه (اثری هنری مانند فیلم، عکس، نقاشی و نمایشنامه) حاصل از نتایج پایان‌نامه/ رساله و تمامی‌طرحهای تحقیقاتی کلیه واحدهای دانشگاه اعم از دانشکده ها، مراکز تحقیقاتی، پژوهشکده ها، پارک علم و فناوری و دیگر واحدها باید با مجوز کتبی صادره از معاونت پژوهشی دانشگاه و براساس آئین نامه های مصوب انجام شود.

ماده ۴ - ثبت اختراع و تدوین دانش فنی و یا ارائه یافته ها در جشنواره‌های ملی، منطقه‌ای و بین‌المللی که حاصل نتایج مستخرج از پایان‌نامه/ رساله و تمامی‌طرح‌های تحقیقاتی دانشگاه باید با هماهنگی استاد راهنما یا مجری طرح از طریق معاونت پژوهشی دانشگاه انجام گیرد.

ماده ۵- این آیین‌نامه در ۵ ماده و یک تبصره در تاریخ۸۷/۴/۱ در شورای پژوهشی و در تاریخ ۸۷/۴/۲۳ در هیأت رئیسه دانشگاه به تایید رسید و در جلسه مورخ ۸۷/۷/۱۵ شورای دانشگاه به تصویب رسیده و از تاریخ تصویب در شورای دانشگاه لازم‌الاجرا است.

اینجانب علیرضا کاویانی‌فر به شماره دانشجویی ... دانشجوی رشته مهندسی کامپیوتر مقطع تحصیلی دکتری دانشکده مهندسی برق و کامپیوتر متعهد می‌شوم کلِیه نکات مندرج در آئین نامه حق مالکیت مادی و معنوی در مورد نتایج پژوهش های علمی‌دانشگاه تربیت مدرس را در انتشار یافته های علمی‌مستخرج از

پایان‌نامه/رساله تحصیلی خود رعایت نمایم. در صورت تخلف از مفاد آئین نامه فوق الاشعار به دانشگاه وکالت و نمایندگی می‌دهم که از طرف اینجانب نسبت به لغو امتیاز اختراع بنام بنده و یا هر گونه امتیاز دیگر و تغییر آن به نام دانشگاه اقدام نماید. ضمناً نسبت به جبران فوری ضرر و زیان حاصله بر اساس برآورد دانشگاه اقدام خواهم نمود و بدینوسیله حق هر گونه اعتراض را از خود سلب نمودم»

تاریخ و امضا:

آیین نامه چاپ پایان‌نامه (رساله)های دانشجویان دانشگاه تربیت مدرس

نظر به اینکه چاپ و انتشار پایان نامه (رساله)های تحصیلی دانشجویان دانشگاه تربیت مدرس، مبین بخشی از فعالیتهای علمی- پژوهشی دانشگاه است بنابراین به منظور آگاهی و رعایت حقوق دانشگاه،دانش آموختگان این دانشگاه نسبت به رعایت موارد ذیل متعهد می‌شوند: ماده ۱: در صورت اقدام به چاپ پایان نامه (رساله)ی خود، مراتب را قبلاً به‌طور کتبی به «دفتر نشر آثارعلمی» دانشگاه اطلاع دهد. ماده ۲: در صفحه سوم کتاب (پس از برگ شناسنامه) عبارت ذیل را چاپ کند: «کتاب حاضر، حاصل پایان نامه کارشناسی ارشد/ رساله دکتری نگارنده در رشته مهندسی کامپیوتر است که در بهمن ۱۴۰۴ در دانشکده‌مهندسی برق و کامپیوتر دانشگاه تربیت مدرس به راهنمایی سرکار خانم/جناب آقای دکتر ...، مشاوره سرکار خانم/جناب آقای دکتر ... از آن دفاع شده است.» ماده ۳: به منظور جبران بخشی از هزینه‌های انتشارات دانشگاه، تعداد یک درصد شمارگان کتاب (در هر نوبت‌چاپ) را به «دفتر نشر آثارعلمی» دانشگاه اهدا کند. دانشگاه می‌تواند مازاد نیاز خود را به نفع مرکز نشر درمعرض فروش قرار دهد. ماده ۴: در صورت عدم رعایت ماده ۳، ۵۰% بهای شمارگان چاپ شده را به‌عنوان خسارت به دانشگاه تربیت‌مدرس، تأدیه کند. ماده ۵: دانشجو تعهد و قبول می‌کند در صورت خودداری از پرداخت بهای خسارت، دانشگاه می‌تواند خسارت مذکور را از طریق مراجع قضایی مطالبه و وصول کند؛ به علاوه به دانشگاه حق می‌دهد به منظور استیفای حقوق خود، از طریق دادگاه، معادل وجه مذکور در ماده ۴ را از محل توقیف کتابهای عرضه شده نگارنده برای فروش، تامین نماید. ماده ۶: اینجانب علیرضا کاویانی‌فر دانشجوی رشته مهندسی کامپیوتر مقطع دکتری تعهد فوق وضمانت اجرایی آن را قبول کرده، به آن ملتزم می‌شوم.

استاد راهنما: دکتر ...

تاریخ:

امضا:

تقدیم

(متن تقدیم به خانواده، اساتید و ...)

قدردانی

(متن تشکر و قدردانی از اساتید راهنما، مشاور و دوستان)

علیرضا کاویانی‌فر

بهمن ۱۴۰۴

چکیده

سیستم‌های نرم‌افزاری معاصر به طور فزاینده‌ای در محیط‌های پویا و همراه با عدم قطعیت عمل می‌کنند. این عدم قطعیت‌ها که ناشی از عواملی مانند تداخل شبکه، نوسانات بار ترافیکی یا تغییرات محیطی هستند، می‌توانند به طور جدی اهداف کیفی سیستم را به خطر اندازند. سیستم‌های خودوفقی (Self-Adaptive Systems) با بهره‌گیری از حلقه‌های بازخوردی، راهکاری بنیادین برای مقابله با این چالش ارائه می‌دهند. با این حال، چالش اصلی زمانی بروز می‌کند که این سیستم‌ها با فضای وفقی وسیع (شامل صدها یا هزاران گزینه پیکربندی) مواجه می‌شوند. روش‌های موجود، از جمله روش‌های مبتنی بر یادگیری ماشین، جستجو، و حتی یادگیری تقویتی کلاسیک، در چنین فضاهایی با مشکلاتی نظیر نیاز به داده‌های برچسب‌دار، عدم مقیاس‌پذیری، و مهم‌تر از همه، گرفتار شدن در بهینه‌های محلی و عدم توانایی تعمیم به شرایط محیطی جدید مواجه هستند. هدف اصلی این رساله، توسعه و اعتبارسنجی روشی نوین مبتنی بر یادگیری تقویتی عمیق با مکانیسم شکل‌دهی تصادفی پاداش (Randomized Reward-Shaped Deep Reinforcement Learning – RS-DRL) برای مدیریت کارآمد عدم قطعیت در این سیستم‌ها است. روش پیشنهادی، هسته یادگیری خود را بر پایه شبکه‌های کیو_عمیق (DQN) استوار ساخته و آن را درون چارچوب استاندارد مدیریت خودمختار MAPE-K یکپارچه می‌سازد. در این معماری، سیگنال محیطی با نتایج حاصل از بازبینی صوری توسط ابزار UPPAAL-SMC ترکیب شده تا عامل یادگیرنده تحت هدایت محدودیت‌های صوری آموزش ببیند. نوآوری اصلی رساله، مکانیسم «شکل‌دهی تصادفی پاداش» است که طی آن، زیرمجموعه‌ای تصادفی از تجربیات ناموفق گذشته با تخصیص پاداش خوش‌بینانه بازطراحی می‌شوند. اگرچه پیکربندی مدل صوری نیازمند دانش دامنه است، اما مکانیسم شکل‌دهی تصادفی پیشنهادی به خودی خود مستقل از دامنه بوده و نیاز به دانش صریح از اهداف (Domain Knowledge) برای اکتشاف ندارد. ارزیابی تجربی بر روی مطالعه موردی DeltaIoT (در شبکه‌های حسگر بی‌سیم) با فضاهای وفقی تا ۴۰۹۶ گزینه، نشان‌دهنده برتری قاطع روش RS-DRL نسبت به روش‌های پایه از جمله DQN، HER، Soft HER و DLASER+ است. به عنوان مثال، در هدف آستانه‌ای TT، روش پیشنهادی موفق به کاهش نرخ اتلاف بسته تا ۲۲/۲۳٪ و کاهش تأخیر تا ۰۷/۳۹٪ در مقایسه با روش DLASER+ به عنوان قوی‌ترین رقیب پژوهش گردید. تحلیل‌های آماری معنادار صحت و استواری نتایج را در شرایط پویای محیطی تأیید می‌کنند. یافته‌های این رساله نشان می‌دهد که رویکرد RS-DRL یک استراتژی بهینه‌سازی مقاوم و خودوفق برای نسل بعدی سیستم‌های نرم‌افزاری پیچیده است.

واژگان کلیدی: سیستم‌های خودوفقی، یادگیری تقویتی عمیق، شکل‌دهی تصادفی پاداش، مدیریت عدم قطعیت، بازبینی صوری، UPPAAL-SMC، فضاهای وفقی بزرگ، اینترنت اشیاء، شبکه‌های حسگر بی‌سیم

فهرست مطالب

تقدیم ث

فهرست جداول ز

فهرست تصاویر ژ

فهرست نشانه‌ها و اختصارات س

فصل ۱: کلیات ۱

۱-۱ مقدمه ۱

۱-۱-۱ بحران پیچیدگی در سیستم‌های نرم‌افزاری معاصر ۱

۱-۱-۲ ناکارآمدی رویکردهای سنتی در عمل ۲

۱-۱-۳ ظهور سیستم‌های خودوفق به عنوان یک راه حل ضروری ۲

۱-۲ اهداف رساله ۲

۱-۳ چالش‌ها ۳

۱-۴ مرور پیشینه و شکاف تحقیقاتی ۴

۱-۴-۱ دسته‌بندی رویکردهای موجود ۵

۱-۴-۲ محدودیت‌های کلیدی رویکردهای موجود ۵

۱-۴-۳ تبیین صریح شکاف تحقیقاتی ۶

۱-۵ بیان دقیق‌تر مسئله با فرمول‌بندی ریاضی مختصر ۶

۱-۵-۱ تعریف فضای وفقی ۶

۱-۵-۲ مدل فرآیند تصمیم‌گیری مارکوف ۷

۱-۵-۳ تابع هدف ۷

۱-۵-۴ قیود عملیاتی مسئله ۷

۱-۶ بیان مسئله ۸

۱-۷ سوالات تحقیق ۸
۱-۸ محدوده و قلمرو پژوهش ۱۰
۱-۸-۱ موضوعات داخل در محدوده پژوهش ۱۰
۱-۸-۲ موضوعات خارج از محدوده پژوهش ۱۰
۱-۹ مخاطبان و کاربردهای بالقوه ۱۱
۱-۱۰ روش پیشنهادی ۱۱
۱-۱۰-۱ مرور کلی بر راهکار RS-DRL ۱۱
۱-۱۰-۲ یکپارچه‌سازی با حلقه کنترلی MAPE-K ۱۱
۱-۱۱ نوآوری‌های اصلی رساله ۱۲
۱-۱۲ مروری بر فصول رساله ۱۳

فصل ۲: مبانی نظری و مدل‌سازی سیستم ۱۵

۲-۱ مقدمه ۱۵
۲-۲ مبانی سیستم‌های خودوفقی ۱۵
۲-۲-۱ مفاهیم اصلی و الگوهای معماری ۱۵
۲-۲-۲ چالش‌های عدم قطعیت در زمان اجرا ۱۶
۲-۳ مدل‌سازی عدم قطعیت در سیستم‌های خودوفقی ۱۷
۲-۳-۱ منابع عدم قطعیت ۱۷
۲-۳-۲ رویکردهای مدل‌سازی در زمان اجرا ۱۷
۲-۳-۳ مطالعه موردی انگیزشی: سیستم DeltaIoT ۱۷
۲-۳-۳-۱ مرور کلی سیستم و منابع عدم قطعیت ۱۸
۲-۳-۳-۲ فضای وفقی و بازنمایی پیکربندی ۱۸
۲-۴ یادگیری تقویتی برای وفقی‌سازی ۱۹
۲-۴-۱ اصول یادگیری تقویتی (MDP، یادگیری Q) ۱۹
۲-۴-۲ یادگیری تقویتی عمیق (DQN) برای فضاهای بزرگ ۱۹
۲-۴-۳ مفهوم یادگیری مداوم در حلقه‌های وفقی ۲۰
۲-۴-۴ مدل صوری برای تصمیم‌گیری تحت عدم قطعیت (ویژه DeltaIoT) ۲۰
۲-۴-۴-۱ تعریف فضای حالت ۲۱
۲-۴-۴-۲ فضای عمل: گزینه‌های وفقی گسسته ۲۱

۲-۴-۴-۳ فرمول‌بندی تابع پاداش ۲۱
۲-۵ بیان مسئله: یادگیری سیاست وفقی بهینه با عملیات مداوم ۲۲
۲-۶ خلاصه فصل ۲۳

فصل ۳: مرور پیشینه پژوهش و کارهای مرتبط ۲۴
۳-۱ مقدمه و روش‌شناسی مرور ۲۴
۳-۲ مبانی نظری: عدم‌قطعیت و تصمیم‌گیری در سیستم‌های خودوفق ۲۵
۳-۳ رویکردهای اصلی مدیریت فضای وفقی در زمان اجرا ۲۶
۳-۳-۱ هرس فضای وفقی مبتنی بر یادگیری ماشین سنتی ۲۷
۳-۳-۲ روش‌های مبتنی بر جستجوی آنلاین و استراتژی‌های اکتشافی ۲۷
۳-۳-۳ یادگیری تقویتی زمان‌اجرا و کنترل خودمختار ۲۸
۳-۴ تکنیک‌های پیشرفته یادگیری تقویتی: کاوش، شکل‌دهی پاداش و نگاه به گذشته ۲۹
۳-۴-۱ چالش‌های استراتژی کاوش در فضاهای وفقی بزرگ ۲۹
۳-۴-۲ ناکارآمدی مکانیزم بازپخش تجربه با نگاه به گذشته (HER) در محیط‌های آستانه‌محور ۳۰
۳-۴-۳ مبانی نظری و ریاضیاتی شکل‌دهی پاداش ۳۱
۳-۵ ارزیابی و تایید صوری در فرآیند خودوفق‌سازی ۳۲
۳-۵-۱ تأیید صوری کمّی زمان اجرا (QVR) ۳۳
۳-۵-۲ چارچوب‌های ActivFORMS و ابزار UPPAAL-SMC ۳۳
۳-۵-۳ معماری‌های ترکیبی یادگیری و ارزیابی صوری (مدل‌های ترادف) ۳۴
۳-۶ سنتز: شکاف تحقیقاتی سه‌گانه و موقعیت‌دهی علمی RS-DRL ۳۵
۳-۷ خلاصه فصل ۳۵

فصل ۴: روش پیشنهادی ۳۶
۴-۱ اصول معماری و سازماندهی فرآیند اجرایی ۳۶
۴-۲ فرآیند یکپارچه خودوفقی: حلقه سطح سیستم ۳۸
۴-۳ مسیر تصمیم‌گیری آنلاین: یک پیاده‌روی در معماری ۴۰
۴-۳-۱ گام اول: پایش و ساخت وضعیت (Monitor) ۴۰
۴-۳-۲ گام دوم: تحلیل و تشخیص ضرورت تطبیق (Analyze) ۴۰
۴-۳-۳ گام سوم: برنامه‌ریزی و استنتاج تطبیقی (Plan) ۴۱

۴-۳-۴ گام چهارم: اعتبارسنجی، بازخورد و اجرا (Execute به همراه Knowledge) . . ۴۱
۴-۴ معماری داخلی RS-DRL: تفکیک دغدغه‌ها در فرآیند یادگیری ۴۱
۴-۵ مکانیزم هسته: شکل‌دهی تصادفی پاداش ۴۳
۴-۶ مدیریت دانش و هماهنگی جریانات ناهمگام ۴۶
۴-۷ پوشش پرسش‌های پژوهش ۴۷
۴-۸ خلاصه فصل ۴۷

فصل ۵: ارزیابی تجربی ۴۹
۵-۱ مقدمه ۴۹
۵-۲ اهداف ارزیابی و سوالات تحقیق ۴۹
۵-۳ تنظیمات آزمایشگاهی ۵۰
۵-۳-۱ نمونه‌های مطالعه موردی ۵۰
۵-۳-۲ مجموعه داده‌ها ۵۱
۵-۳-۳ پلتفرم پیاده‌سازی ۵۱
۵-۳-۴ تنظیمات فراپارامترها (Hyperparameters) ۵۱
۵-۴ روش‌های پایه و مقایسه ۵۱
۵-۵ معیارهای ارزیابی ۵۲
۵-۵-۱ معیارهای عملکرد یادگیری عامل هوشمند ۵۲
۵-۵-۲ معیارهای عملکرد سیستم ۵۳
۵-۶ نتایج و تحلیل‌ها ۵۳
۵-۶-۱ منحنی‌های یادگیری و رفتار همگرایی ۵۳
۵-۶-۲ مقایسه عملکرد به ازای هر هدف ۵۵
۵-۶-۳ ارزیابی مقایسه‌ای در برابر تمام روش‌های پایه ۵۶
۵-۶-۴ مطالعه ابلیشن: حساسیت به فاکتور شکل‌دهی ρ ۵۷
۵-۶-۵ تحلیل مقیاس‌پذیری ۵۷
۵-۷ اعتبارسنجی آماری ۵۸
۵-۸ خلاصه فصل ۵۹

فصل ۶: بحث، نتیجه‌گیری و پیشنهادها ۶۰
۶-۱ بحث در مورد یافته‌های کلیدی ۶۰

۶-۱-۱ نظریه‌پردازی مکانیزم: چرا RS-DRL کار می‌کند؟ ۶۰
۶-۱-۲ تحلیل تعاملات و مصالحه‌ها (به‌عنوان مثال، تاخیر در مقابل از دست رفتن بسته) ۶۱
۶-۱-۳ پیامدهای مهندسی سیستم‌های خودوفقی ۶۲
۶-۲ محدودیت‌ها و تهدیدات اعتبار (Threats to Validity) ۶۳
۶-۲-۱ اعتبار داخلی (Internal Validity) ۶۳
۶-۲-۲ اعتبار خارجی (External Validity) ۶۳
۶-۲-۳ اعتبار سازه (Construct Validity) ۶۴
۶-۳ مسیرهایی برای تحقیقات آینده ۶۵
۶-۳-۱ استراتژی‌های انطباقی و وفقی شکل‌دهی ρ ۶۵
۶-۳-۲ گسترش به دامنه‌های پیوسته و حساس به ایمنی ۶۵
۶-۳-۳ اعتبارسنجی بین‌دامنه و تحلیل‌های نظری ۶۶
۶-۴ نتیجه‌گیری ۶۷
۶-۴-۱ خلاصه بیان مسئله و راهکار پیشنهادی ۶۷
۶-۴-۲ مرور مشارکت‌ها و شواهد تجربی ۶۷
۶-۴-۳ سخنان پایانی ۶۸

کتاب‌نامه **۷۰**

فصل آ: جریان‌های کاری کامل فرآیند **۷۳**

فصل ب: لیست کامل الگوریتم‌ها **۷۴**

فصل پ: نتایج تجربی گسترده **۷۵**

فصل ت: مصنوعات پیاده‌سازی و پکیج بازتولید **۷۶**

فهرست جداول

۴-۱ طبقه‌بندی زمانی فرآیندهای معماری RS-DRL ۳۸
۴-۲ قراردادهای تعاملی و همگام‌سازی جریانات آنلاین و ناهمگام ۴۷

۵-۱ هم‌راستایی اهداف ارزیابی با سوالات تحقیق ۵۰
۵-۲ نمونه‌های مورد مطالعه سیستم DeltaIoT ۵۰
۵-۳ نتایج اهداف کمینه‌سازی در DeltaIoTv1 (پاداش مجانبی / گام‌ها) ۵۵
۵-۴ نتایج اهداف کمینه‌سازی در DeltaIoTv1.1 . . . . . ۵۵
۵-۵ عملکرد سیستم برای اهداف آستانه‌ای ترکیبی (TTS Goal) ۵۶
۵-۶ عملکرد سیستم برای اهداف آستانه‌ای مطلق (TT Goal) ۵۶
۵-۷ تحلیل مقیاس‌پذیری الگوریتم با بزرگ شدن فضای تطابق ۵۷
۵-۸ مقادیر احتمال (p-values) برای مقایسه‌های معنادار و غیرمعنادار ۵۸

۶-۱ خلاصه‌ای از مشارکت‌های کلیدی و شواهد تجربی ۶۷

فهرست تصاویر

۴-۱ معماری سطح بالای رویکرد RS-DRL در قالب حلقه .MAPE-K این نمودار تعامل میان سیستم مدیریت‌شونده، محیط، و مؤلفه‌های کنترلی را به همراه مخزن دانش (Knowl-edge) مشترک نشان می‌دهد. ۳۷

۴-۲ نمودار فعالیت تفصیلی UML با خطوط شناوری (Swimlanes) برای تفکیک مسئولیت‌های پایش، تحلیل، برنامه‌ریزی و اعتبارسنجی. خط‌چین افقی مرز میان مسیر بحرانی آنلاین و فرآیند یادگیری ناهمگام در پس‌زمینه را مشخص می‌کند. ۳۹

۴-۳ ساختار داخلی مؤلفه برنامه‌ریز RS-DRL در سه مسیر موازی: استنتاج (فقط‌خواندنی)، فیدبک (ترکیب پاداش محیطی و صوری)، و آموزش (به‌روزرسانی ناهمگام با استفاده از شکل‌دهی تصادفی پاداش). ۴۲

۴-۴ منطق عملیاتی مکانیزم شکل‌دهی تصادفی پاداش. این نمودار فرآیند جایگزینی خوش‌بینانه پاداش‌های صفر با مقدار ۱ را بر اساس فاکتور ρ نشان می‌دهد (برچسب‌گذاری مجدد خوش‌بینانه، نه روش‌های مبتنی بر پتانسیل). ۴۳

۴-۵ جریان داده تفصیلی در مخزن دانش. این نمودار نحوه ذخیره‌سازی تجربیات در بافر بازپخش، اعمال مکانیزم شکل‌دهی تصادفی پاداش پیش از محاسبات گرادیان، و به‌روزرسانی سیاست‌های یادگیری در پس‌زمینه را به تصویر می‌کشد. ۴۶

۵-۱ منحنی‌های یادگیری الگوریتم‌ها در محیط DeltaIoTv1 (پاداش در برابر گام‌ها) ۵۴

۵-۲ منحنی‌های یادگیری الگوریتم‌ها در محیط DeltaIoTv1.1 . . . . . ۵۴

۵-۳ تاثیر مقادیر مختلف ضریب شکل‌دهی ρ بر پاداش مجانبی و عملکرد کل ۵۷

فهرست نشانه‌ها و اختصارات

(لیست نمادهای ریاضی مانند ρ، γ و اختصارات مانند DRL، SAS)

فصل ۱

کلیات

۱-۱ مقدمه

در حوزه مهندسی سیستم‌های نرم‌افزاری معاصر، پیچیدگی به‌عنوان یکی از بارزترین ویژگی‌ها خودنمایی می‌کند. محیط عملیاتی پویا نقشی محوری در شکل‌دهی به این پیچیدگی دارد و سیستم‌ها را ناگزیر می‌سازد تا با شرایط نامشخصی که اغلب تنها پس از عملیاتی شدن بروز می‌یابند، مقابله کنند.

۱-۱-۱ بحران پیچیدگی در سیستم‌های نرم‌افزاری معاصر

در سال‌های اخیر، ظهور معماری‌های توزیع‌شده، رایانش ابری، اینترنت اشیاء (IoT) و سیستم‌های فیزیکی–سایبری، ابعاد جدیدی از پیچیدگی محاسباتی و عدم قطعیت‌های زمان اجرا را به همراه داشته است. آمار و گزارش‌های صنعتی نشان می‌دهند که خرابی و توقف غیرمنتظره سیستم‌های نرم‌افزاری در مقیاس صنعتی، هزینه‌های سرسام‌آوری را تحمیل می‌کند. به عنوان مثال، در زیرساخت‌های ابری بزرگ مقیاس، هر دقیقه توقف خدمات (downtime) می‌تواند تا هزاران دلار ضرر مالی مستقیم و آسیب جدی به اعتبار برندها به همراه داشته باشد. از سوی دیگر، در شبکه‌های حسگر بی‌سیم و اینترنت اشیاء صنعتی، افزایش ناگهانی تداخل امواج یا رانش مفهومی در کانال‌های ارتباطی می‌تواند نرخ هدررفت داده‌ها را به شدت افزایش دهد. در چنین سناریوهایی، نرخ خطای قابل قبول بسیار کوچک است و سیستم‌ها باید بلافاصله خود را با محیط تطبیق دهند. این موضوع نشان می‌دهد که مدیریت عدم قطعیت در سیستم‌های نوین، یک چالش دانشگاهی محض نیست، بلکه یک ضرورت عملی و فوری در صنایع مختلف به شمار می‌رود.

۱-۱-۲ ناکارآمدی رویکردهای سنتی در عمل

سیستم‌های نرم‌افزاری سنتی که مبتنی بر پیکربندی‌های ایستا در زمان طراحی یا قواعد ثابت زمان اجرا (نظیر دستورات شرطی ساده «اگر–آنگاه») هستند، فاقد انعطاف‌پذیری لازم برای پاسخ‌گویی به نوسانات پویای محیطی می‌باشند. در این سیستم‌ها، تمامی شرایط عملیاتی و رفتارهای سیستم باید در زمان طراحی پیش‌بینی شوند. با این حال، در محیط‌های پیچیده واقعی، فرضیات زمان طراحی همواره نقض می‌شوند. برای نمونه، در سیستم‌های هوشمند مدیریت ترافیک شهری یا کنترل مسیریابی شبکه‌های نسل پنجم (5G)، تغییرات پویای حجم داده‌ها، نوسان بار ترافیکی و خرابی‌های ناگهانی سخت‌افزاری موجب شکست قوانین ایستای زمان طراحی می‌شوند. سیستم‌هایی که بر قواعد سخت‌گیرانه زمان طراحی تکیه می‌کنند، در سناریوهای خارج از پیش‌بینی‌های اولیه با زوال عملکرد یا توقف کامل مواجه می‌شوند.

۱-۱-۳ ظهور سیستم‌های خودوفق به عنوان یک راه حل ضروری

به منظور غلبه بر این محدودیت‌ها، سیستم‌های خودوفق (SAS) به عنوان یک پارادایم اساسی در مهندسی نرم‌افزار نوین ظهور کردند. این سیستم‌ها مجهز به حلقه‌های بازخوردی خودمختار هستند که به سیستم اجازه می‌دهند تا به طور پیوسته وضعیت درونی و بیرونی خود را پایش کرده، انحرافات از اهداف تعیین‌شده را تحلیل کرده، تغییرات بهینه را برنامه‌ریزی و در نهایت در زمان اجرا اعمال نمایند.

ایده اولیه این رویکرد به کارهای بنیادین شرکت IBM در سال ۲۰۰۳ تحت عنوان رایانش خودمختار توسط کیفارت و چس [۱] بازمی‌گردد. در ادامه، اوریزی و همکاران (۱۹۹۹) [۲] یکی از نخستین رویکردهای معماریمحور برای سیستم‌های خودوفق را معرفی کردند که بر استفاده از مدل‌های زمان اجرا برای هدایت تصمیمات وفقی تأکید داشت. همچنین چنگ و همکاران (۲۰۰۹) [۳] در نقشه‌راه تحقیقاتی مهندسی نرم‌افزار برای سیستم‌های خودوفق، بر اهمیت مدل‌های خودمختار زمان اجرا، مدیریت عدم قطعیت و تضمین‌های کیفیت تأکید نمودند. ادغام حلقه‌های کنترلی خودمختار با تکنیک‌های تصمیم‌گیری هوشمند، سنگ بنای سیستم‌های خودوفق مدرن را شکل می‌دهد.

۱-۲ اهداف رساله

این رساله در راستای تکمیل شکاف‌های تحقیقاتی پیشین [۳] و پاسخ به چالش‌های حل‌نشده، یک هدف اصلی و سه هدف فرعی را دنبال می‌کند:

هدف اصلی

توسعه و اعتبارسنجی روش یادگیری تقویتی عمیق با شکل‌دهی تصادفی پاداش (RS-DRL) به‌گونه‌ای که بر مشکل بنیادین گیرافتادن در بهینه‌های محلی در فضاهای وفقی گسسته و بزرگ غلبه کند. این روش سیستم‌های خودوفقی را قادر می‌سازد تا بدون نیاز به بازآموزی پرهزینه و اخلال‌گر، عملکردی نزدیک به بهینه را به‌طور پیوسته در شرایط عدم قطعیت محیطی حفظ نمایند.

این هدف اصلی، هسته‌ی نوآوری رساله (مکانیزم شکل‌دهی تصادفی پاداش) را مستقیماً به چالش اساسی شناسایی‌شده در بیان مسئله پیوند می‌دهد.

اهداف فرعی

در راستای تحقق هدف اصلی، اهداف فرعی زیر تعریف شده‌اند:

۱. **طراحی معماری مقیاس‌پذیر و یکپارچه:** ایجاد معماری مدولاری که عامل RS-DRL را به‌طور ناهمگام با حلقه‌ی خودمختار MAPE-K ادغام نموده و امکان اعتبارسنجی صوری تصمیمات پیش از اجرا را فراهم آورد. این هدف تضمین می‌کند که راهکار پیشنهادی نه تنها هوشمند، بلکه قابل‌اعتماد و قابل‌استقرار در سیستم‌های زمان‌اجرای واقعی باشد.

۲. **مدیریت کارآمد فضاهای وفقی گسسته با ابعاد بسیار بزرگ:** نشان دادن این که روش RS-DRL قادر است در سیستم‌هایی با فضای وفقی تا ۴۰۹۶ گزینه (مانند DeltaIoTv2)، بدون افت محسوس عملکرد، گزینه‌های بهینه یا نزدیک به بهینه را در زمان کوتاه انتخاب کند. این هدف، مقیاس‌پذیری عملی راهکار را برای کاربردهای دنیای واقعی اثبات می‌نماید.

۳. **ارزیابی تجربی جامع و اثبات برتری آماری:** مقایسه‌ی کمی دقیق روش RS-DRL با روش‌های پیشرفته‌ی موجود (شامل ϵ-greedy DQN، HER، Soft HER، DLASER و ML4EAS) از نظر عملکرد مجانبی، سرعت همگرایی، و بهینه‌سازی چندهدفه، همراه با ارائه‌ی تحلیل‌های آماری معنی‌دار (مانند t-test و فواصل اطمینان) برای تصدیق برتری روش پیشنهادی.

۱-۳ چالش‌ها

چالش‌ها و منابع عدم قطعیت در سیستم‌های مختلف به شکل‌های گوناگونی بروز می‌یابند. به‌عنوان مثال، در سیستم‌های مستقر در محیط‌های پویا نظیر برنامه‌های اینترنت اشیا (IoT) در شهرهای هوشمند، تغییرات

آب و هوایی تأثیر قابل توجهی می‌گذارد. همچنین محیط‌های ابری با تقاضای منابع متغیر، امنیت سایبری با تهدیدات نوظهور و مداوم، تجارت الکترونیک و شبکه‌های اجتماعی با رفتار غیرقابل پیش‌بینی کاربران، و سیستم‌های توزیع‌شده با تأخیر شبکه، خرابی سخت‌افزار و تغییرات زیرساختی روبه‌رو هستند.

هنگامی که سیستم با فضای وفقی گسترده و گزینه‌های متعدد تغییر مواجه می‌شود، انتخاب گزینه‌ای وفقی که با اهداف سیستم همسو باشد، از نظر محاسباتی بسیار پرهزینه و دشوار، و در برخی موارد تقریباً غیرممکن می‌گردد.

روش‌های موجود برای مدیریت عدم قطعیت در سیستم‌های خودوفقی با محدودیت‌های قابل‌توجهی مواجه هستند. به‌عنوان مثال، روش‌های مبتنی بر یادگیری ماشین نیازمند داده‌های آموزشی برچسب‌دار هستند که نمایانگر محیط سیستم باشند. به دست آوردن چنین داده‌هایی در زمان طراحی به دلیل عدم قطعیت فراوان بسیار دشوار است. از طرفی، روش‌های مبتنی بر جستجو (مانند الگوریتم‌های صعود تپه و الگوریتم‌های ژنتیک) با مشکلات مقیاس‌پذیری مواجه هستند و عملکرد آن‌ها با تغییر شرایط محیطی دچار افت محسوس می‌شود.

یادگیری تقویتی عمیق (DRL) علیرغم قدرت بالا، خود با چالش‌های اساسی نظیر گیرافتادن در بهینه‌های محلی مواجه است. این چالش در سیستم‌هایی با فضای وفقی گسسته و بزرگ به‌مراتب تشدید می‌شود؛ زیرا هرچه تعداد گزینه‌های وفقی بیشتر باشد، احتمال همگرایی مدل به راه‌حل‌های بهینه‌ی محلی به جای بهینه‌های سراسری افزایش می‌یابد و کاوش کافی در فضای عمل دشوارتر می‌گردد. بهینه‌های محلی به راه‌حل‌هایی اشاره دارند که تنها در یک منطقه محدود از فضای مسئله بهینه هستند و به صورت سراسری بهترین راه‌حل ممکن محسوب نمی‌شوند. در زمینه DRL، این بدان معناست که مدل ممکن است سیاستی را یاد بگیرد که تحت شرایط خاص به خوبی عملکرد مناسبی دارد، اما در صورت تغییر محیط قادر به تعمیم‌پذیری و حفظ عملکرد مطلوب نیست.

علاوه بر این، آموزش مجدد یا بازآموزی (Retraining) مدل در زمان اجرا فرآیندی پرهزینه است. بازآموزی شامل مراحل متعددی مانند جمع‌آوری داده‌های جدید، ارزیابی مجدد ساختار و پارامترهای مدل، و تست مدل به‌روزشده برای اطمینان از عملکرد بهتر آن نسبت به نسخه قبلی است. در طول این فرآیند، سیستم ممکن است به صورت غیربهینه عمل کند که در سیستم‌های حیاتی، این دوره عملکرد نامطلوب می‌تواند پیامدهای جبران‌ناپذیری به همراه داشته باشد.

۱-۴ مرور پیشینه و شکاف تحقیقاتی

اگرچه کارهای کلاسیک در حوزه سیستم‌های خودوفق زیربنای نظری این پارادایم را ایجاد کرده‌اند، اما انتخاب استراتژی تطبیق بهینه در زمان اجرا همچنان یک چالش اساسی است. در این بخش، خلاصه‌ای از پیشینه

پژوهش در طراحی مکانیزم‌های تصمیم‌گیری در فضاهای وفقی بزرگ ارائه شده و شکاف‌های تحقیقاتی موجود شناسایی می‌گردد.

۱-۴-۱ دسته‌بندی رویکردهای موجود

به طور کلی، تحقیقات صورت گرفته در ادبیات سیستم‌های خودوفق جهت هدایت فرآیند تصمیم‌گیری زمان اجرا را می‌توان به سه دسته اصلی تقسیم‌بندی نمود:

۱. **رویکردهای مبتنی بر یادگیری ماشین سنتی:** این رویکردها از مدل‌های یادگیری نظارت‌شده یا بدون نظارت برای کاهش ابعاد فضای وفقی یا پیش‌بینی اثرات پیکربندی‌ها استفاده می‌کنند.

۲. **رویکردهای مبتنی بر جستجو:** این روش‌ها مسائل تطبیق زمان اجرا را به عنوان یک مسئله بهینه‌سازی مدل‌سازی کرده و از الگوریتم‌های اکتشافی یا جستجوی تصادفی استفاده می‌کنند.

۳. **رویکردهای مبتنی بر یادگیری تقویتی:** این مدل‌ها به عامل اجازه می‌دهند تا سیاست بهینه تطبیق را از طریق آزمون و خطا در تعامل مستقیم با محیط یاد بگیرد.

۱-۴-۲ محدودیت‌های کلیدی رویکردهای موجود

هر یک از دسته‌بندی‌های فوق در مواجهه با چالش‌های زمان اجرا دارای محدودیت‌های خاصی هستند:

- **محدودیت‌های روش‌های مبتنی بر یادگیری ماشین:** کویین و همکاران در پژوهش‌های خود [۴، ۵] نشان داده‌اند که این روش‌ها وابستگی شدیدی به در دسترس بودن داده‌های آموزشی برچسب‌دار با کیفیت بالا در زمان طراحی دارند. در محیط‌های به شدت نامشخص، توزیع داده‌های محیطی زمان اجرا با توزیع فرضی زمان طراحی تفاوت دارد که منجر به پدیده رانش مفهومی (Concept Drift) [۶] و در نتیجه افت شدید دقت پیش‌بینی مدل‌ها می‌گردد.

- **محدودیت‌های روش‌های مبتنی بر جستجو:** کوکر (۲۰۱۵) [۷] و کینیر و همکاران (۲۰۱۸، ۲۰۲۱) [۸، ۹] نشان داده‌اند که اگرچه الگوریتم‌های بهینه‌سازی تصادفی کارایی مناسبی در فضاهای وفقی کوچک دارند، اما در فضاهای وفقی گسسته و بزرگ (با صدها یا هزاران پیکربندی) دچار مشکل مقیاس‌پذیری و تأخیر محاسباتی بسیار بالا در زمان اجرا می‌شوند. در چنین فضاهایی، اجرای جستجو در زمان تطبیق با محدودیت‌های زمانی شدید سیستم همخوانی ندارد.

- **محدودیت‌های روش‌های یادگیری تقویتی سنتی:** رویکردهای مبتنی بر یادگیری تقویتی عمیق سنتی (نظیر الگوریتم‌های DQN با مکانیزم‌های کاوش متداول مانند ϵ-greedy) به دلیل محدودیت کاوشگری خود، در فضاهای بزرگ به شدت مستعد گیرافتادن در بهینه‌های محلی هستند [۱۰، ۱۱]. عبور از این نقاط بهینه محلی معمولاً نیازمند فرآیندهای پرهزینه و زمان‌بر بازآموزی (Retraining) در زمان اجرا است که موقتاً ثبات عملکرد سیستم را مختل می‌سازد.

۱-۴-۳ تبیین صریح شکاف تحقیقاتی

با بررسی ادبیات موجود، شکاف تحقیقاتی کلیدی زیر شناسایی می‌شود:

«هیچ‌یک از رویکردهای موجود در مهندسی سیستم‌های خودوفق قادر نبوده‌اند به طور همزمان (۱) بدون نیاز به بازآموزی‌های محاسباتی مکرر و اخلال‌گر، از بهینه‌های محلی در زمان اجرا عبور کنند، (۲) در فضاهای وفقی گسسته و بزرگ (تا مقیاس ۴۰۹۶ گزینه تغییر) به صورت زمان‌حقیقی مقیاس‌پذیر باشند، و (۳) مستقل از دانش دامنه خاص (Domain-agnostic) برای هدایت فرآیند کاوش عمل کنند.»

این رساله با هدف پر کردن این خلاء تحقیقاتی، روش نوین یادگیری تقویتی عمیق با شکل‌دهی تصادفی پاداش (RS-DRL) را پیشنهاد و اعتبارسنجی می‌کند.

۱-۵ بیان دقیق‌تر مسئله با فرمول‌بندی ریاضی مختصر

برای درک عمیق‌تر چالش‌های تصمیم‌گیری در سیستم‌های خودوفق، در این بخش مسئله تطبیق زمان اجرا را به صورت ریاضی فرمول‌بندی می‌کنیم. این فرمول‌بندی سطح بالا مبنایی برای ارائه جزئیات کامل در فصل دوم خواهد بود.

۱-۵-۱ تعریف فضای وفقی

فضای وفقی سیستم شامل مجموعه‌ای متناهی از گزینه‌های پیکربندی یا اعمال وفقی گسسته است که با $\mathcal{A}$ نشان داده می‌شود:

$$\mathcal{A} = \{a_۱, a_۲, \ldots, a_n\}, \quad n \leq ۴۰۹۶ \tag{۱-۱}$$

هر گزینه $a_i \in \mathcal{A}$ نمایانگر یک پیکربندی شبکه، توزیع منابع، یا پهنای باند اختصاص‌یافته در یک لحظه مشخص است. در این تحقیق، ابعاد فضای وفقی تا مقیاس بزرگ ($n = ۴۰۹۶$) مورد بررسی قرار گرفته

است.

۱-۵-۲ مدل فرآیند تصمیم‌گیری مارکوف

مسئله تطبیق را به صورت یک فرآیند تصمیم‌گیری مارکوف (MDP) مدل‌سازی می‌کنیم [۱۲، ۱۳]. فضای حالت $\mathcal{S}$ وضعیت‌های سیستم و شرایط تداخل محیطی را پوشش می‌دهد. در هر گام زمانی t، عامل وفقی بر اساس سیاست تطبیق خود $\pi(a|s)$، اقدامی مانند $a_t \in \mathcal{A}$ را در حالت $s_t \in \mathcal{S}$ انتخاب می‌کند که منجر به انتقال به حالت بعدی s_{t+1} و دریافت پاداش بازخوردی $R(s_t, a_t)$ می‌گردد.

۱-۵-۳ تابع هدف

هدف نهایی عامل، پیدا کردن یک سیاست بهینه تطبیق π^* است که امید ریاضی مجموع پاداش‌های تنزیل‌شده بلندمدت سیستم را بیشینه کند:

$$\max_{\pi} \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t)\right] \tag{۱-۲}$$

که در آن $\gamma \in [0, 1)$ نرخ تنزیل پاداش‌های آتی است.

۱-۵-۴ قیود عملیاتی مسئله

سیاست استخراج‌شده π^* باید تحت سه قید عملیاتی و بنیادین زیر بهینه‌سازی شود:

۱. **قید خروج از بهینه‌های محلی بدون بازآموزی:** عامل یادگیرنده باید قادر باشد بدون ایجاد اخلال در سیستم در قالب بازآموزی مکرر در زمان اجرا، از نقاط بهینه محلی عبور کند.

۲. **قید مقیاس‌پذیری زمانی:** زمان پاسخ محاسباتی برای انتخاب عمل بهینه $a^* \in \mathcal{A}$ در زمان اجرا باید بسیار ناچیز و در حد میلی‌ثانیه باشد تا به سرعت به رانش مفهومی پاسخ داده شود.

۳. **قید استقلال از دانش دامنه:** مکانیزم کاوشگری عامل نباید به فرمول‌بندی پاداش‌های دقیق و متکی بر دانش تخصصی مهندسان دامنه وابسته باشد و باید به طور خودمختار رفتار کاوشگری بهینه را هدایت کند.

۱-۶ بیان مسئله

سیستم‌های خودوفقی مدرن در محیط‌های پویا و غیرقابل‌پیش‌بینی نظیر شبکه‌های اینترنت اشیاء، رایانش ابری و شهرهای هوشمند عمل می‌کنند. این سیستم‌ها اغلب با فضاهای وفقی گسسته و بزرگ مواجه هستند که می‌تواند شامل صدها یا هزاران گزینه پیکربندی مختلف باشد (برای مثال، شبکه حسگر DeltaIoTv2 دارای ۴۰۹۶ گزینه وفقی است).

مدل‌های سنتی یادگیری تقویتی عمیق (DRL) که برای تصمیم‌گیری در این فضاها به کار گرفته می‌شوند، با دو چالش بنیادین مواجه‌اند:

۱. **گیرافتادن در بهینه‌های محلی:** الگوریتم‌هایی مانند ϵ-greedy DQN به دلیل ماهیت کاوشگری محدود خود، اغلب در نقاط بهینه محلی متوقف می‌شوند و قادر به کشف سیاست‌های سراسری بهینه نیستند [۱۴]. این مشکل در فضاهای وفقی بزرگ به مراتب حادتر می‌شود.

۲. **نیاز به بازآموزی مکرر و پرهزینه:** در مواجهه با تغییرات محیطی (رانش مفهومی [۶])، مدل‌های سنتی نیاز به بازآموزی کامل دارند. این فرآیند زمان‌بر، محاسباتی سنگین است و در طول دوره بازآموزی، سیستم عملکرد نامطلوبی از خود نشان می‌دهد که در سیستم‌های حیاتی می‌تواند پیامدهای جبران‌ناپذیری به همراه داشته باشد.

روش‌های پیشرفته‌تری مانند HER و Soft HER نیز اگرچه در حوزه محیط‌های هدف‌محور موفق بوده‌اند، اما برای سیستم‌های خودوفقی با اهداف مبهم و آستانه‌ای (مانند بهینه‌سازی همزمان مصرف انرژی، تأخیر و تلفات بسته) طراحی نشده‌اند و نمی‌توانند بدون دانش دامنه خاص عمل کنند.

بنابراین، مسئله اصلی این رساله را می‌توان به صورت زیر فرموله کرد:

طراحی روش یادگیری تقویتی عمیق که (۱) بدون نیاز به بازآموزی مکرر و پرهزینه، از بهینه‌های محلی در فضاهای وفقی گسسته و بزرگ (تا ۴۰۹۶ گزینه) عبور کند، (۲) قابلیت تعمیم‌پذیری به شرایط محیطی متغیر را داشته باشد، و (۳) مستقل از دانش دامنه خاص عمل کند.

۱-۷ سوالات تحقیق

با توجه به محدودیت‌های رویکردهای پیشین و نیز دستاوردهای آن‌ها که در بخش مرور پیشینه و شکاف تحقیقاتی تبیین گردید [۳، ۶]، این رساله در صدد پاسخ به پرسش‌های اساسی زیر همراه با زیرسوالات متناظر است:

۱. **سوال اول:** چگونه می‌توان روشی مبتنی بر یادگیری تقویتی عمیق طراحی کرد که به‌طور مؤثر از بهینه‌های محلی در فضاهای وفقی گسسته و بزرگ، بدون نیاز به بازآموزی مکرر و پرهزینه، عبور کند؟

- **زیرسوال ۱-۱:** چه مکانیزمی برای تزریق هوشمند و کنترل‌شده‌ی «خوش‌بینی» به فرآیند بازپخش تجربیات یادگیرنده بدون تأثیر نامطلوب بر پایداری مدل می‌توان طراحی کرد؟
- **زیرسوال ۱-۲:** چگونه می‌توان از تجربیات ناموفق گذشته عامل بدون وارد شدن به فاز آموزش مجدد زمان اجرا استفاده کرد؟
- **زیرسوال ۱-۳:** چه تضمینی برای عدم همگرایی واگرا یا بی‌ثبات در محیط‌های دارای تداخل شدید در فرآیند یادگیری وجود دارد؟

۲. **سوال دوم:** چگونه می‌توان مکانیزم‌های شکل‌دهی تصادفی پاداش (Random Reward Shaping) را با بازپخش تجربه (Experience Replay) ادغام کرد تا تعمیم‌پذیری سیستم در تغییرات شرایط محیطی بهبود یابد؟

۳. **سوال سوم:** چگونه روش پیشنهادی می‌تواند فضاهای وفقی گسسته و بزرگ (از ۲۱۶ تا ۴۰۹۶ گزینه) را در حین حفظ عملکردی نزدیک به بهینه، به شکل کارآمدی مدیریت کند؟

۴. **سوال چهارم:** تا چه حد روش پیشنهادی (RS-DRL) از روش‌های موجود نظیر یادگیری ای‌گرایانه حریصانه (ϵ-greedy DQN)، HER، Soft HER، DLASER و ML4EAS از منظر عملکرد مجانبی، سرعت همگرایی، و بهینه‌سازی چندهدفه، برتری و عملکرد بهتری از خود نشان می‌دهد؟

- **زیرسوال ۴-۱:** عملکرد مجانبی RS-DRL در بلندمدت در مقایسه با الگوریتم‌های استاندارد یادگیری تقویتی نظیر DQN و مدل‌های پس‌نگر نظیر HER در محیط‌های غیرایستا چگونه است؟
- **زیرسوال ۴-۲:** سرعت همگرایی روش پیشنهادی در فضاهای وفقی با ابعاد بسیار بزرگ (مانند شبکه با ۴۰۹۶ حالت پیکربندی) چه تفاوتی با رویکردهای یادگیری عمیق فشرده‌کننده نظیر DLASER+ دارد؟
- **زیرسوال ۴-۳:** آیا برتری‌های عملکردی حاصل از روش پیشنهادی از نظر آماری در مقایسه با سایر استراتژی‌ها معنادار است؟

۱-۸ محدوده و قلمرو پژوهش

برای تبیین دقیق مرزها و قلمرو این رساله دکتری، دامنه‌ی مسائل تحت پوشش و موضوعاتی که خارج از اهداف این کار پژوهشی هستند، به صورت زیر مشخص می‌گردد:

۱-۸-۱ موضوعات داخل در محدوده پژوهش

- **فضاهای وفقی گسسته:** این رساله به طور خاص بر سیستم‌های خودوفقی با فضاهای وفقی گسسته متمرکز است که پیکربندی‌های سیستم شامل گزینه‌های عددی یا ساختاری متمایز و مشخص هستند.
- **یادگیری تقویتی مدل آزاد:** رویکرد یادگیری تقویتی به کار گرفته شده از نوع مدل آزاد (Model-free) و مبتنی بر معماری شبکه‌های کیو عمیق (DQN) است که به هیچ مدل از پیش‌تعریف‌شده‌ای از رفتار محیط وابسته نیست.
- **سیستم‌های حسگر بی‌سیم صنعتی:** مطالعه موردی مرجع و اصلی جهت شبیه‌سازی و ارزیابی تجربی، بستر نام‌آشنای DeltaIoT (در نسخه‌های ۱v و ۲v با ابعاد مختلف فضای وفقی) است که چالش‌های محیطی شدید و پویایی بالایی دارد.

۱-۸-۲ موضوعات خارج از محدوده پژوهش

- **فضاهای وفقی پیوسته:** تحلیل و مدل‌سازی سیستم‌های خودوفق با پارامترهای تطبیق کاملاً پیوسته خارج از محدوده این کار بوده و به عنوان زمینه‌های پیشنهادی برای تحقیقات آینده در نظر گرفته شده است.
- **سیستم‌های چندعاملی:** تمرکز این رساله بر سناریوهای تصمیم‌گیری تک‌عاملی متمرکز است که در آن یک عامل خودمختار مرکزی مسئولیت تطبیق را بر عهده دارد. بررسی سیستم‌های توزیع‌شده با یادگیری تقویتی چندعاملی (MARL) مد نظر نبوده است.
- **اثبات قضایای ریاضی همگرایی محض:** تضمین همگرایی و کارایی روش پیشنهادی به صورت تجربی و مبتنی بر شبیه‌سازی‌های فشرده و گسترده و تحلیل‌های آماری معتبر اثبات شده است، نه از طریق روابط اثبات تئوریک همگرایی ریاضی صوری در شرایط نامحدود.

۱-۹ مخاطبان و کاربردهای بالقوه

دستاوردها و راهکارهای ارائه شده در این رساله دکتری می‌تواند برای طیف وسیعی از محققان و حوزه‌های کاربردی سودمند واقع شود:

- **محققان مهندسی نرم‌افزار:** فعالان پژوهشی در حوزه‌های مهندسی نرم‌افزار خودوفق، حلقه‌های کنترلی خودمختار (MAPE-K) و تضمین کیفیت زمان اجرا.
- **طراحان سیستم‌های اینترنت اشیاء و شبکه‌های حسگر بی‌سیم:** مهندسان و معماران شبکه‌های صنعتی هوشمند که نیازمند استقرار سیستم‌های به شدت پایدار و بهینه در مواجهه با اختلالات پویای سیگنال هستند.
- **توسعه‌دهندگان الگوریتم‌های یادگیری تقویتی عمیق:** طراحان الگوریتم‌های هوش مصنوعی که با چالش گیرافتادن در بهینه‌های محلی در مسائل کاربردی با فضای عمل گسسته مواجه هستند.

۱-۱۰ روش پیشنهادی

۱-۱۰-۱ مرور کلی بر راهکار RS-DRL

راهکار پیشنهادی در این رساله مبتنی بر توسعه یک معماری یادگیری هوشمند است که به جای تکیه بر دانش ایستا، از طریق تعامل مستقیم با محیط، سیاست‌های بهینه را استخراج می‌کند. این روش مسئله تطبیق را به‌عنوان یک فرآیند تصمیم‌گیری مارکوف (MDP) مدل‌سازی کرده و با ادغام مدل‌های عمیق و لایه مدیریت خودمختار، یک چرخه پیوسته از یادگیری و اجرا را فراهم می‌آورد. هسته نوآورانه این روش، مکانیزم «شکل‌دهی تصادفی پاداش» است که با بازطراحی تصادفی برخی تجربیات ناموفق گذشته، عامل را قادر به فرار از بهینه‌های محلی بدون نیاز به بازآموزی پرهزینه می‌سازد. تمرکز اصلی بر ارائه چارچوبی است که در آن، عامل یادگیرنده نه تنها بر اساس پاداش‌های آنی، بلکه با هدف حفظ عملکرد مطلوب در بازه زمانی طولانی، کاهش نیاز به بازآموزی‌های مکرر و پایداری معیارهای کیفی سیستم نظیر مصرف انرژی و زمان تأخیر، عمل کند.

۱-۱۰-۲ یکپارچه‌سازی با حلقه کنترلی MAPE-K

روش پیشنهادی با بهره‌گیری از چارچوب استاندارد MAPE-K، فرآیند تطبیق را به صورت ساختاریافته مدیریت می‌کند. در این معماری، عامل RS-DRL در مؤلفه برنامه‌ریزی (Plan) مستقر شده و خروجی‌های آن (گزینه وفقی پیشنهادی) پیش از اجرا توسط ابزار تأیید صوری UPPAAL-SMC از نظر ایمنی و قابلیت اطمینان ارزیابی

می‌گردند. سپس گزینه‌های تأییدشده بر روی سیستم مدیریت‌شونده اعمال می‌شوند. این یکپارچگی تضمین می‌کند که تصمیمات هوشمندانه‌ی عامل DRL تنها پس از اعتبارسنجی صوری به اجرا درآیند.

۱-۱۱ نوآوری‌های اصلی رساله

بنیانِ علمی و سهم پژوهشی این پایان‌نامه در قالب چهار نوآوری اصلی زیر تبیین می‌گردد که هر یک به طور مستقیم به رفع یکی از محدودیت‌های کلیدی در ادبیات موضوع می‌پردازد:

۱. **نوآوری اول: الگوریتم یادگیری تقویتی با شکل‌دهی تصادفی پاداش (RS-DRL):** ارائه یک مکانیزم نوین شکل‌دهی پاداش تصادفی زمان اجرا که برای اولین بار برخی از تجربیات ناموفق گذشته را بدون نیاز به دانش دامنه خاص بازطراحی می‌کند. در حالی که رویکردهای پیشین مانند HER نیازمند تعریف اهداف صریح زمان اجرا و دانش عمیق فرمول‌بندی محیط هستند، روش پیشنهادی با تخصیص پاداش مثبت به زیرمجموعه‌ای تصادفی از تجربیات گذشته، بدون هیچ‌گونه دانش دامنه‌ای، عامل یادگیرنده را قادر به کاوش کارآمد فضای وفقی و خروج موفقیت‌آمیز از بهینه‌های محلی می‌سازد. این رویکرد به ویژه مانع همگرایی واگرا یا گیرافتادن در نقاط پایدار محلی می‌شود.

۲. **نوآوری دوم: ادغام معماری RS-DRL در حلقه MAPE-K:** طراحی یک معماری سیستماتیک و تلفیقی ناهمگام که عامل یادگیرنده هوشمند را در مؤلفه تصمیم‌گیری حلقه کنترل خودمختار تعبیه می‌کند. برخلاف رویکردهای کلاسیک که فرآیند آموزش عمیق را به صورت آفلاین یا کاملاً مجزا انجام می‌دهند، در این معماری فرآیند آموزش پیوسته و استنتاج به صورت پویا با پایش وضعیت فیزیکی سیستم همگام شده و گزینه‌های تصمیم‌گیری پیش از اعمال، برای اطمینان صد درصدی از پایداری کیفیت خدمات، از فیلتر اعتبارسنجی صوری ابزار UPPAAL-SMC عبور می‌کنند.

۳. **نوآوری سوم: مدیریت فضای وفقی مقیاس‌پذیر و گسسته در ابعاد بزرگ:** ارائه یک راهکار مقیاس‌پذیر زمان‌اجرا جهت اتخاذ تصمیمات وفقی چندهدفه در ابعاد تا ۴۰۹۶ گزینه تغییر. در حالی که روش‌های سنتی مبتنی بر جستجو نظیر الگوریتم ژنتیک یا روش‌های شبیه‌سازی تصادفی به علت سربار محاسباتی بسیار بالا در مواجهه با ابعاد ۴۰۹۶ با اخلال مواجه می‌شوند، روش پیشنهادی با کاهش زمان تصمیم‌گیری استنتاج مدل به مقیاس میلی‌ثانیه، پایداری همزمان مصرف انرژی و قابلیت اطمینان ارتباطات را تضمین می‌کند.

۴. **نوآوری چهارم: معماری جداسازی زمانی (Temporal Separation):** توسعه یک متدولوژی سازماندهی زمانی که مسیر تصمیم‌گیری آنلاین (با استنتاج زیر میلی‌ثانیه) را به طور کامل از ریسمان‌های سنگین

آموزش ناهمگام در پس‌زمینه تفکیک می‌کند. این نوآوری معماری تضمین می‌کند که افزایش ابعاد فضای وفقی تا ۴۰۹۶ گزینه، کوچک‌ترین خللی در سرعت تصمیم‌گیری بلادرنگ و پایداری عملیاتی سیستم وارد نسازد.

۱-۱۲ مروری بر فصول رساله

ساختار رساله به گونه‌ای طراحی شده است که به‌طور نظام‌مند روند توسعه، اجرا و ارزیابی رویکرد پیشنهادی را در اختیار خواننده قرار دهد. در ادامه، نقش و کارکرد هر یک از فصول در راستای زنجیره استدلالی و پاسخ به سوالات رساله تشریح می‌گردد:

فصل ۱: کلیات

در این فصل، کلیات پژوهش، بیان مسئله، اهداف و نوآوری‌های رساله به همراه سوالات تحقیق و مروری بر ساختار کل رساله ارائه می‌گردد تا خواننده با بستر کلی و ضرورت طرح پژوهش آشنا شود.

فصل ۲: مبانی نظری و مدل سیستم

این فصل زیربنای ریاضی و مفهومی لازم برای درک روش پیشنهادی را فراهم می‌کند. در این فصل، مبانی سیستم‌های خودوفقی (حلقه MAPE-K)، منابع عدم قطعیت و رویکردهای مدل‌سازی زمان اجرا (از جمله ابزار UPPAAL-SMC)، و اصول یادگیری تقویتی (فرآیند تصمیم‌گیری مارکوف و الگوریتم DQN) تشریح شده و مدل ریاضی سیستم بستر DeltaIoT فرموله می‌گردد.

فصل ۳: مرور ادبیات و کارهای مرتبط

با بهره‌گیری از چارچوب نظری فصل دوم، این فصل به بررسی انتقادی و جامع رویکردهای پیشین پرداخته و با تحلیل نقاط قوت و ضعف آن‌ها، ضرورت علمی و موقعیت رساله حاضر در بدنه دانش سیستم‌های خودوفق را مستحکم می‌سازد.

فصل ۴: روش پیشنهادی RS-DRL: معماری و الگوریتم

این فصل هسته نوآوری و طراحی فنی رساله را پوشش می‌دهد. در این فصل، جزئیات الگوریتمی مکانیزم شکل‌دهی تصادفی پاداش، معماری یکپارچه ناهمگام با حلقه کنترلی خودمختار و فرآیند اعتبارسنجی صوری پیش از اجرا توسط ابزار UPPAAL-SMC به تفصیل ارائه می‌شود تا به سوالات اول و دوم رساله پاسخ داده شود.

فصل ۵: ارزیابی تجربی

این فصل جنبه‌های عملیاتی، پیاده‌سازی و اعتبارسنجی تجربی رساله را شامل می‌شود. شبیه‌سازی‌ها و مقایسه‌های گسترده با رقبای نام‌آشنا روی سناریوهای مختلف بستر DeltaIoT مستند شده و تحلیل‌های آماری

معتبر در این فصل ارائه می‌شود تا پاسخ به سوالات سوم و چهارم رساله تصدیق گردد.

فصل ۶: بحث و نتیجه‌گیری

در نهایت، این فصل مصالحه‌های رفتاری، محدودیت‌های فنی و دستاوردهای کاربردی سیستم را جمع‌بندی نموده و افق‌های جدیدی برای پژوهش‌های آتی در توسعه مهندسی سیستم‌های خودوفق هوشمند ترسیم می‌کند.

فصل ۲

مبانی نظری و مدل‌سازی سیستم

۲-۱ مقدمه

هدف این فصل، ارائه‌ی چارچوب نظری مستحکمی‌است که درک صحیح راهکار پیشنهادی RS-DRL را ممکن می‌سازد. سیستم‌های نرم‌افزاری مدرن همواره با عدم قطعیت‌های مداوم مواجه هستند و باید در زمان اجرا به‌طور پیوسته ساختار و رفتار خود را با شرایط در حال تغییر وفق دهند. این فصل در ابتدا اصول اولیه سیستم‌های خودوفقی و مؤلفه‌های آن را تشریح می‌کند، سپس منابع عدم قطعیت و روش‌های مدل‌سازی آن را معرفی می‌نماید. در ادامه، با معرفی سیستم DeltaIoT به‌عنوان یک مطالعه موردی عینی، مبانی یادگیری تقویتی و الگوریتم DQN را در قالب این سیستم فرموله کرده و در نهایت مسئله اصلی رساله را به صورت صوری بیان می‌کنیم.

۲-۲ مبانی سیستم‌های خودوفقی

۲-۲-۱ مفاهیم اصلی و الگوهای معماری

سیستم‌های خودوفقی (Self-Adaptive Systems) یک رویکرد برجسته و مؤثر برای مقابله با عدم قطعیت به شمار می‌روند. ایده کلیدی در این سیستم‌ها، تجهیز آن‌ها به یک یا چند حلقه بازخوردی کنترلی است که به صورت مداوم مؤلفه‌های داخلی سیستم و محیط خارجی آن را پایش کرده و در صورت نیاز سیستم را با تغییرات منطبق می‌نمایند [۱].

مهمترین و شناخته‌شده‌ترین الگوی معماری برای پیاده‌سازی این سیستم‌ها، حلقه خودمختار MAPE-K است. این حلقه در کنار سیستم هدف و منطق کسب‌وکار، جریان کنترلی زیر را اعمال می‌کند:

- **پایش (Monitor):** سنسورها وضعیت محیط فیزیکی، زیرساخت و خود سیستم را به صورت زمان‌حقیقی استخراج می‌کنند.
- **تحلیل (Analyze):** داده‌های پایش‌شده را پردازش کرده و مشخص می‌کند که آیا انحرافی از اهداف کیفی و کمی سیستم رخ داده است و نیاز به اعمال تغییر وجود دارد یا خیر.
- **برنامه‌ریزی (Plan):** در صورت نیاز تطبیق، راهکارها و استراتژی‌های موجود (گزینه‌های وفقی) مقایسه شده و گزینه بهینه برای شرایط فعلی برگزیده می‌شود.
- **اجرا (Execute):** استراتژی برنامه‌ریزی‌شده در قالب یک سری از دستورات اجرایی از طریق عملگرها (Actuators) بر روی سیستم اعمال می‌گردد.
- **دانش (Knowledge):** به‌عنوان حافظه مرکزی عمل کرده و اطلاعات، مدل‌ها و داده‌های بین چهار مرحله قبلی را هماهنگ و نگهداری می‌کند. راهکار پیشنهادی ما (RS-DRL) مستقیماً به‌عنوان هسته اصلی بخش برنامه‌ریزی در این الگو فعالیت می‌کند.

۲-۲-۲ چالش‌های عدم قطعیت در زمان اجرا

عدم قطعیت در سیستم‌های خودوفقی، به هرگونه انحراف از دانش قطعی اطلاق می‌شود که می‌تواند باعث کاهش اطمینان نسبت به تصمیمات زمان اجرا گردد [۳].

به عنوان نمونه‌های عینی، می‌توان به موارد گوناگونی در دامنه‌های مختلف اشاره کرد: در برنامه‌های مبتنی بر اینترنت اشیاء (IoT) که در محیط‌های شهری پراکنده هستند، تغییرات مداوم و غیرقابل پیش‌بینی آب و هوا منجر به افت کیفیت سیگنال و اختلال در ارسال اطلاعات سنسورها می‌شود. در محیط‌های ابری، نوسان شدید در تقاضای منابع از سوی کاربران، و در حوزه‌های امنیتی، ظهور بی‌وقفه و تصادفی حملات سایبری از مصادیق برجسته عدم قطعیت هستند.

اهمیت و چالش‌برانگیز بودن عدم قطعیت از آنجا نشأت می‌گیرد که مفروضات، پیکربندی‌ها و تصمیمات در نظر گرفته شده در فاز طراحی معماری نرم‌افزار را بی‌اعتبار می‌سازد. در نتیجه، سیستم باید قادر باشد با یادگیری آنلاین و پویا در زمان اجرا، اهداف اصلی خود را محقق نماید؛ وگرنه با نقض کیفیت خدمات صدمات سنگین احتمالی را تجربه خواهد نمود.

۲-۳ مدل‌سازی عدم قطعیت در سیستم‌های خودوفقی

۲-۳-۱ منابع عدم قطعیت

منابع عدم قطعیت به‌طور کلی در سه دسته وسیع طبقه‌بندی می‌شوند: نخست، نامشخص بودن موجودیت‌های محیطی نظیر پارازیت در حسگرها، نویزها و تغییرات جوی که از کنترل ساختار درونی سیستم خارج هستند. دوم، ماهیت غیرقابل اتکای منابع خود سیستم مثل نقص در قطعات زیربنایی، کاهش سریع سطح باتری یا مشکلات نرم‌افزاری تعاملات میان سرویس‌ها؛ و در نهایت عدم قطعیت در اهداف متغیر و ترجیحات نامشخص ذینفعان در محیط‌های تجاری.

۲-۳-۲ رویکردهای مدل‌سازی در زمان اجرا

جهت درک و محاسبه رفتار سیستم در هنگام بروز مقادیر ناشناخته، استفاده از مدل‌سازی‌های زمان اجرا (Run-time Models) ضروری به نظر می‌رسد. در شرایطی که فضای جستجو بسیار بزرگ و مخاطره تصمیم‌گیری به دلیل شرایط مبهم بالاست، رویکرد پیشنهادی از مکانیزم‌های مدل‌سازی ریاضی صوری بهره‌گیری و پشتیبانی می‌کند.

در این راستا ابزارهایی مانند موتور UPPAAL-SMC (Statistical Model Checking — وارسی مدل آماری) و چارچوب ActivFORMS به کار گرفته می‌شوند. این زیرساخت‌ها با استفاده از اتوماتای زمان‌دار (Timed Automata) مدلی رفتاری از سیستم هدف و محیط آن را توسعه داده و از طریق روش‌های وارسی مدل آماری می‌توانند به سرعت گزینه‌های وفقی پیشنهاد شده توسط سیستم هوشمند را پیش از اجرا در سیستم فیزیکی ارزیابی، تایید یا رد کنند. در واقع، این روش با برآوردِ آماریِ احتمال نقض اهداف کیفی خدمات (QoS)، امنیت تصمیمات را تضمین می‌کند. این قابلیت نقشی کلیدی در امنیت راهکار پیشنهادی (در بخش تاییدگر گزینه وفقی) این رساله ایفا می‌کند.

۲-۳-۳ مطالعه موردی انگیزشی: سیستم DeltaIoT

برای عینی‌تر ساختن این مفاهیم انتزاعی در یک محیط عملیاتی، در ادامه سیستم DeltaIoT را معرفی می‌کنیم؛ یک شبکه حسگر بی‌سیم که به‌عنوان یک معیار استاندارد، چالش‌های اصلی فضاهای وفقی بزرگ و عدم قطعیت‌های محیطی را به خوبی به تصویر می‌کشد.

۲-۳-۳-۱ مرور کلی سیستم و منابع عدم قطعیت

سیستم DeltaIoT یک شبکه حسگر بی‌سیم (WSN) با تعاملات پیچیده است که به صورت گسترده به‌عنوان یک محک استاندارد (Benchmark) برای ارزیابی سیستم‌های خودوفقی به کار می‌رود [۱۵]. این شبکه بسته به نسخه مورد استفاده، مابین ۱۵ تا ۳۷ گره (سنسور) دارد. دو منبع اصلی عدم قطعیت در این سیستم در زمان اجرا بروز می‌یابند:

۱. **نویز و تداخل شبکه (Network Interference/Noise):** تغییرات محیطی و مداخلات پیرامونی که موجب نوسانات پیش‌بینی‌نشده در کیفیت ارتباطات از 40dB- تا 15dB+ می‌گردد.

۲. **تغییرپذیری بار پیام‌ها (Message Load Variability):** نوسان در تعداد پیام‌های تولید شده توسط هر سنسور (بین ۰ تا ۱۰ پیام) با الگوهای انتقال متفاوت در طول زمان.

همانطور که در مطالعات پیشین (و روند نشان داده شده در شکل ۲ مقاله مرجع) نمایان است، این عدم قطعیت‌ها باعث شکل‌گیری افت شدید عملکرد سیستم در طی چرخه‌های وفقی گشته و اهمیت حیاتی مکانیسم‌های تطابق پویا را برجسته می‌کنند.

۲-۳-۳-۲ فضای وفقی و بازنمایی پیکربندی

فضای وفقی (Adaptation Space) دربرگیرنده مجموعه تمامی‌گزینه‌ها و پیکربندی‌های ممکنی است که سیستم می‌تواند در زمان اجرا اتخاذ نماید. در DeltaIoT، سه نسخه متفاوت با مشخصاتی به شرح زیر طراحی گردیده است [۱۵]:

- **DeltaIoTv1:** شامل ۱۵ سنسور و دارای ۲۱۶ گزینه وفقی متمایز.
- **DeltaIoTv1.1:** شامل ۱۶ سنسور و دربرگیرنده ۱۲۹۶ گزینه وفقی است.
- **DeltaIoTv2:** نسخه وسیع‌تر این شبکه متشکل از ۳۷ سنسور که دارای فضای عظیم متشکل از ۴۰۹۶ تصمیم ممکن می‌باشد. توجه داشته باشید که ابعاد این فضا از ترکیب سطوح مختلف توان انتقال و توزیع مسیرهای شبکه حاصل شده است.

هریک از این گزینه‌های وفقی در حقیقت ترکیبی از پارامترهای مختلف هستند؛ نظیر سطوح توان انتقال انرژی (Transmission Power) برای سنسورهای مشخص و توزیع نسبی مسیرهای شبکه‌ای (Link Distribution). انتخاب یک ترکیب از این پارامترها، یک پیکربندی کاملاً جدید و واحد را شکل می‌دهد.

۲-۴ یادگیری تقویتی برای وفقی‌سازی

۲-۴-۱ اصول یادگیری تقویتی (MDP، یادگیری Q)

یادگیری تقویتی از طریق تعامل مستمر یک عامل (Agent) با محیط اطرافش بر پایه آزمون و خطا بنا شده است و این محیط به لحاظ ریاضی از طریق فرآیند تصمیم‌گیری مارکوف (MDP) فرموله می‌شود. فرآیند MDP خود شامل فضای حالت (State)، فضای عمل (Action) و توابع پاداش (Reward) است.

فرآیند تصمیم‌گیری مارکوف، برای هر حالت خاص، عامل عمل مشخصی را اتخاذ می‌کند و مطابق با معادله مشهور بلمن (Bellman Equation) محیط ارزش عمل اتخاذ شده را در قالب یک پاداش ارزیابی می‌نماید و به وضعیت بعدی منتقل می‌شود. این مفهوم در سیستم‌های خودوفقی مانند محیط DeltaIoT به این صورت اعمال می‌گردد که وضعیت سیستم با معیارهایی نظیر انرژی ذخیره شده سیستم و تاخیر اندازه‌گیری شده فضای حالت را تشکیل می‌دهند و اقدام عامل، همان گزینه وفقی مورد انتخاب خواهد بود و محاسبه پاداش نیز طبق معادله بهینه‌سازی سیستم حاصل می‌گردد. نقش الگوریتم مشهور یادگیری Q ایجاد جدولی جامع برای تخصیص و پیدا کردن ارزش‌هاست تا مدل به سیاست بهینه همگرا شود.

۲-۴-۲ یادگیری تقویتی عمیق (DQN) برای فضاهای بزرگ

اما زمانی که با سیستم‌های واقعی که دارای فضای عمل گسسته ولی بسیار پهناور (برای مثال ۴۰۹۶ گزینه وفقی در شبکه‌های اینترنت اشیاء) مواجه هستیم، استفاده از متد کلاسیک یادگیری Q با چالش نفرین ابعاد (Curse of Dimensionality) روبه‌رو می‌شود؛ زیرا ذخیره و بروزرسانی جدولی با میلیون‌ها حالت و عمل از لحاظ محاسباتی ناکارآمد و در بسیاری موارد غیرممکن است.

شبکه‌های Q عمیق (DQN) راهکاری هوشمندانه برای این محدودیت ارائه می‌دهند؛ آن‌ها به جای جداول، تابع ارزش کیفیت Q را با استفاده از شبکه‌های عصبی عمیق تقریب می‌زنند. برای غلبه بر بی‌ثباتی آموزش شبکه‌های عصبی در RL، مدل DQN از دو نوآوری برجسته بهره می‌برد: نخست استفاده از مکانیسم حافظه بازپخش تجربه (Experience Replay) که نمونه‌های آموزشی تصادفی‌سازی شده و باعث شکستن همبستگی زمانی متوالی می‌گردد؛ و دوم تعبیه هدف ثابت به واسطه شبکه هدف ثابت (Target Network) که بهبود تثبیت توابع هزینه را در هنگام محاسبه پیش‌بینی‌ها به دنبال دارد.

علیرغم موفقیت DQN در فضاهای بزرگ، این الگوریتم‌ها به طور ذاتی مستعد گیر افتادن در بهینه‌های محلی هستند، به ویژه زمانی که محیط غیرایستا (non-stationary) باشد. دلیل اصلی آن استراتژی کاوش ϵ-greedy است که پس از همگرایی اولیه، کاوش مؤثر را متوقف می‌کند. در سیستم‌های خودوفقی با فضای

وفقی بزرگ، این مشکل حادتر می‌شود، زیرا تغییرات محیطی می‌توانند سیاست بهینه را جابهجا کنند، در حالی که عامل همچنان در سیاست قدیمی گیر افتاده است. روش پیشنهادی این رساله (RS-DRL) در فصل ۴ با معرفی مکانیزم **شکل‌دهی تصادفی پاداش** به این چالش می‌پردازد.

۲-۴-۳ مفهوم یادگیری مداوم در حلقه‌های وفقی

اصلی‌ترین کاربرد این رساله، ادغام مفهوم آموزش پیوسته و مداوم توسط عامل DQN در چارچوب معماری خودوفقی و به‌طور خاص درون حلقه MAPE-K (همانطور که در فصل‌های آتی به تفصیل نشان داده خواهد شد) است. پس از جاگذاری در مؤلفه برنامه‌ریزی (Planner)، عامل یادگیرنده وضعیتی را به‌عنوان ورودی فضای حالت از ماژول پایش دریافت نموده، استراتژی اقدام را برنامه‌ریزی کرده و بر اساس میزان موفقیت انتخابش از سوی مؤلفه تحلیل پاداش می‌گیرد. این ساختار تعاملی، سیستم خودوفقی را از پیکربندی ایستا به ساختار یادگیری پویا تبدیل می‌سازد. مفهوم یادگیری مداوم در اینجا به معنای به‌روزرسانی ناهمگام (Asynchronous) شبکه عصبی در پس‌زمینه است؛ به طوری که عامل همزمان با عملکرد سیستم، از تجربیات جدید برای بهبود تدریجی سیاست خود بهره می‌برد.

در فصل ۳، روش‌های موجود شکل‌دهی پاداش (مانند HER) مرور خواهند شد، و در فصل ۴، نوآوری این رساله یعنی **شکل‌دهی تصادفی پاداش** به عنوان راهکاری برای غلبه بر بهینه‌های محلی معرفی می‌گردد.

۲-۴-۴ مدل صوری برای تصمیم‌گیری تحت عدم قطعیت (ویژه DeltaIoT)

با تبیین مبانی نظری فرآیند تصمیم‌گیری مارکوف و شبکه‌های DQN، اکنون این چارچوب را برای سیستم DeltaIoT پیاده‌سازی می‌کنیم. در ادامه، فضای حالت، فضای عمل و توابع پاداش اختصاصی که در طول این رساله مورد استفاده قرار می‌گیرند، تعریف خواهند شد.

برای امکان‌پذیر کردن یادگیری تقویتی، نیاز به فرمول‌بندی مسأله تطابق شبکه در قالب یک فرآیند تصمیم‌گیری مارکوف (MDP) داریم که نمایانگر چهارگانه (S, A, R, P) است [۱۲، ۱۳]. در اینجا P نمایش‌دهنده تابع احتمال انتقال است؛ هرچند در روش‌های یادگیری تقویتی مدل‌آزاد مانند DQN، عامل بدون نیاز به دانستن صریح P و مستقیماً از طریق تعامل با محیط یاد می‌گیرد.

۲-۴-۴-۱ تعریف فضای حالت

فضای حالت مدل، وضعیت فعلی و پویای سیستم را با اندازه‌گیری سه معیار حیاتی عملکرد بازتاب می‌دهد. به بیان رسمی:

$$S = \{(E, P, L) \mid E \in \mathbb{R}, P \in [۰, ۱۰۰], L \in [۰, ۱۰۰]\}$$

که در آن:

- **انرژی مصرفی (E):** مقدار حقیقیِ نمایانگر انرژی کل مصرفی توسط تمام سنسورهای شبکه در چرخه پیشین است.
- **تلفات بسته‌های شبکه (P):** درصدی از پیام‌های تولید شده که در طول مسیر به مقصد (دروازه اصلی) از دست رفته‌اند (بین ۰ تا ۱۰۰ درصد).
- **تاخیر ارسال (L):** معیاری معادل زمان مورد نیاز جهت رسیدن اطلاعات از سنسورها به گیت‌وی (حداکثر بازه ۱۰۰).

۲-۴-۴-۲ فضای عمل: گزینه‌های وفقی گسسته

فضای عمل شامل تمامی‌استراتژی‌ها و پیکربندی‌های ممکن به صورت مجموعه‌ای از گزینه‌های گسسته تعریف می‌شود:

$$A = \{a_۱, a_۲, \ldots, a_n\}$$

مقدار n بسته به ابعاد شبکه DeltaIoT برابر با ۲۱۶، ۱۲۹۶ یا ۴۰۹۶ در نظر گرفته می‌شود. هر عمل a_i نمایانگر یک پیکربندی مشخص از پارامترهای $\{p_{i۱}, p_{i۲}, \ldots, p_{im}\}$ است.

۲-۴-۴-۳ فرمول‌بندی تابع پاداش

بر اساس ماهیت سیستم و اهداف، از فرمول‌بندی‌های زیر استفاده می‌شود:

پاداش چندهدفه:

$$R = w_E \cdot \left(\frac{E_{max} - E}{E_{max} - E_{min}}\right) + w_P \cdot \left(\frac{P_{max} - P}{P_{max} - P_{min}}\right) + w_L \cdot \left(\frac{L_{max} - L}{L_{max} - L_{min}}\right)$$

وزن‌ها (w_i) مقادیر مثبتی هستند که معمولاً به گونه‌ای انتخاب می‌شوند که مجموع آن‌ها برابر یک باشد ($\sum w_i = ۱$) تا مقدار کل پاداش در بازه [۰, ۱] باقی بماند. مقادیر min و max نیز کران‌های عملیاتی هر معیار هستند که از داده‌های شبیه‌سازی استخراج شده‌اند.

پاداش تک‌هدفه:

$$R = \left(\frac{E_{max} - E}{E_{max} - E_{min}} \right) \cdot w_E$$

در پاداش تک‌هدفه، تنها هدف بیشینه‌سازی بهره‌وری مصرف انرژی است و ضریب w_E معمولاً برابر ۱ در نظر گرفته می‌شود.

اهداف آستانه‌ای (TTS و TT): تخصیص مقادیر پاداش مشروط به صورت دودویی:

$$R = \begin{cases} ۱ & \text{اگر } P < \theta_P \text{ و } L < \theta_L \\ ۰ & \text{در غیر این صورت} \end{cases}$$

که در آن θ_P و θ_L به ترتیب آستانه‌های پذیرش برای تلفات بسته و تاخیر هستند؛ برای مثال در ارزیابی‌های فصل ۵، مقادیر ۱۰٪ و ۵۰ میلی‌ثانیه منظور شده‌اند.

۲-۵ بیان مسئله: یادگیری سیاست وفقی بهینه با عملیات مداوم

پس از تعریف دقیق مؤلفه‌های MDP، اکنون می‌توانیم مسئله اصلی بهینه‌سازی را که روش پیشنهادی RS-DRL به دنبال حل آن است، به صورت صوری بیان کنیم. مسأله نهایی ما یادگیری سیاست وفقی بهینه $\pi^* : S \to A$ است که ارزش پاداش‌های تجمعی تنزیل‌یافته را حداکثر نماید:

$$\pi^* = \arg\max_{\pi} \mathbb{E} \left[\sum_{t=۰}^{\infty} \gamma^t R(s_t, a_t) \mid \pi \right]$$

که در آن $\gamma \in [۰, ۱)$ ضریب تنزیل (Discount Factor) است و میزان اهمیت پاداش‌های آتی را نسبت به پاداش لحظه‌ای مشخص می‌کند.

با این حال، مسئله اصلی این رساله فراتر از بهینه‌سازی استاندارد است و باید سه قید عملیاتی را برآورده سازد:

۱. **مقابله با بهینه‌های محلی:** سیاست وفقی باید توانایی خروج از بهینه‌های محلی را بدون بازآموزی کامل داشته باشد، چرا که محیط به طور مداوم در حال تغییر است (non-stationary).

۲. **فضای عمل گسسته و بسیار بزرگ:** تعداد گزینه‌های وفقی می‌تواند تا ۴۰۹۶ حالت باشد (مانند DeltaIoTv2) که روش‌های جدول‌محور را غیرعملی می‌سازد.

۳. **عدم وابستگی به دانش دامنه:** بر خلاف روش‌های مبتنی بر هدف (مانند HER)، راهکار پیشنهادی نباید به اهداف صریح یا آستانه‌های از پیش تعریف‌شده وابسته باشد.

بنابراین، مسئله نهایی یافتن سیاستی است که نه تنها پاداش تجمعی را بیشینه کند، بلکه در شرایط فوق نیز پایدار و کارا باقی بماند.

این سیاست باید همواره به‌طور مداوم در زمان اجرا عمل کرده و با چالش‌های عدم قطعیت محیطی، فضای عمل گسسته گسترده و پویایی غیرایستا (تغییر شرایط از یک چرخه به چرخه بعدی) مقابله نماید.

۲-۶ خلاصه فصل

این فصل چارچوب نظری و مدل‌سازی مورد نیاز برای درک رویکرد پیشنهادی را ترسیم نمود. ما با اصول سیستم‌های خودوفقی و الگوی MAPE-K آغاز کردیم و سپس منابع عدم قطعیت و روش‌های مدل‌سازی زمان اجرا را بررسی نمودیم. با استفاده از DeltaIoT به‌عنوان یک مطالعه موردی عینی، مسئله وفقی‌سازی را در قالب فرآیند تصمیم‌گیری مارکوف (MDP) فرموله کردیم که شامل تعریف دقیق فضای حالت (انرژی، تلفات، تاخیر)، فضای عمل (تا ۴۰۹۶ گزینه) و توابع پاداش چندهدفه بود. در نهایت، هدف بهینه‌سازی را به‌عنوان یادگیری یک سیاست وفقی برای عملیات مداوم در زمان اجرا بیان کردیم. این زیربنای صوری و عملیاتی، امکان نقد و بررسی دقیق‌تر کارهای پیشین در فصل ۳ و ارائه متدولوژی نوآورانه RS-DRL در فصل‌های آتی را فراهم می‌سازد.

فصل ۳

مرور پیشینه پژوهش و کارهای مرتبط

۳-۱ مقدمه و روش‌شناسی مرور

هدف بنیادین این فصل، ارائه یک بررسی نظام‌مند، انتقادی و عمیق از ادبیات علمی و رویکردهای موجود در زمینه مدیریت عدم قطعیت و تصمیم‌گیری زمان اجرا در سیستم‌های خودوفق (Self-Adaptive Systems - SAS) است. در طراحی مهندسی سیستم‌های خودوفق مدرن، اتخاذ تصمیمات وفقی بهینه در مواجهه با شرایط پویای محیطی و فضاهای تصمیم‌گیری بسیار بزرگ، یکی از چالش‌های بنیادین به شمار می‌رود. کارهای پیشین هر یک از زاویه‌ای خاص به این مسئله نگریسته‌اند؛ با این حال، به منظور درک دقیق سهم علمی این رساله و تبیین دقیق ضرورت ارائه الگوریتم RS-DRL، تحلیل موشکافانه و مقایسه‌ای این رویکردها امری حیاتی است.

بدین منظور، در این فصل روش‌شناسی مروری ساختاریافته‌ای اتخاذ گردیده است. فرآیند شناسایی و گزینش مقالات از طریق جستجوی جامع در پایگاه‌های داده معتبر بین‌المللی از جمله IEEE Xplore، ACM Digital Library، Scopus و Google Scholar صورت پذیرفته است. بازه زمانی جستجو متمرکز بر پژوهش‌های برجسته دو دهه اخیر بوده و کلمات کلیدی نظیر «سیستم‌های خودوفق»، «یادگیری تقویتی آنلاین»، «شکل‌دهی پاداش»، «فضای وفقی بزرگ» و «تأیید صوری زمان اجرا» به عنوان معیارهای اصلی فیلترینگ استفاده شده‌اند. معیارهای ورود مقالات به چرخه بررسی انتقادی شامل تمرکز بر روی مدیریت فضاهای وفقی گسسته در زمان اجرا و توانایی سیستم در تصمیم‌گیری آنلاین تحت شرایط عدم قطعیت بوده است.

جهت ارزیابی نظام‌مند و مقایسه‌ی عادلانه پیشینه پژوهش، یک چارچوب ارزیابی پنج‌بعدی توسعه داده شده است که تمامی آثار بررسی‌شده بر اساس این شاخص‌ها مورد نقد قرار می‌گیرند:

۱. **ابعاد فضای وفقی (Adaptation Space Size):** توانایی متدولوژی پیشنهادی در مقیاس‌پذیری و مدیریت فضاهای تصمیم‌گیری گسسته و بزرگ (فراتر از ۱۰۰۰ گزینه وفقی).

۲. **مدیریت پویایی و عدم‌قطعیت زمان‌اجرا (Handling of Non-Stationarity):** پایداری عملکرد سیستم در مواجهه با رانش مفهومی (Concept Drift) و تغییرات پیش‌بینی‌نشده محیطی بدون افت کیفیت خدمات.

۳. **عدم وابستگی به دانش دامنه خاص (Need for Domain Knowledge):** میزان نیاز به تخصص انسانی زمان توسعه یا طراحی اهداف صریح و پیچیده برای عامل یادگیرنده.

۴. **مقیاس‌پذیری و کارایی محاسباتی (Real-Time Scalability):** توانایی تصمیم‌گیری آنلاین در محدوده میلی‌ثانیه به منظور رعایت قیود پایداری زمانی سیستم.

۵. **قابلیت خروج از بهینه‌های محلی (Ability to Escape Local Optima):** کارایی مکانیزم‌های کاوش در فرار از تله‌های بهینگی محلی در طول یادگیری آنلاین.

در ادامه‌ی این فصل، ابتدا مبانی نظری عدم قطعیت و تصمیم‌گیری در سیستم‌های خودوفق تشریح شده، سپس سه پارادایم اصلی تصمیم‌گیری (مبتنی بر یادگیری ماشین سنتی، مبتنی بر جستجو و مبتنی بر یادگیری تقویتی عمیق) به صورت موشکافانه تحلیل می‌گردند. در نهایت، با بررسی روش‌های پیشرفته بازپخش تجربه و شکل‌دهی پاداش و نقش روش‌های صوری، خلاءهای پژوهشی موجود استخراج و جایگاه الگوریتم پیشنهادی RS-DRL تبیین می‌گردد.

۳-۲ مبانی نظری: عدم‌قطعیت و تصمیم‌گیری در سیستم‌های خودوفق

سیستم‌های خودوفق به منظور حفظ اهداف کیفی خود (نظیر قابلیت اطمینان، مصرف انرژی و تأخیر)، نیازمند پایش مستمر محیط فیزیکی و تصمیم‌گیری بهینه در زمان اجرا هستند [۱]. با این حال، تصمیم‌گیری در این سیستم‌ها تحت تأثیر شدید منابع گوناگون عدم قطعیت قرار دارد. عدم قطعیت در ادبیات مهندسی نرم‌افزار خودوفق [۳] به طور کلی به سه دسته اصلی تقسیم می‌شود: عدم قطعیت محیطی (مانند نوسانات تداخل سیگنال در شبکه‌های اینترنت اشیا)، عدم قطعیت سیستمی (نظیر خرابی‌های ناگهانی سخت‌افزار یا تغییرات زیرساختی) و عدم قطعیت اهداف (تغییر در اولویت‌های کیفی کاربران در طول زمان).

برای مدل‌سازی فرآیند تصمیم‌گیری در مواجهه با این عدم قطعیت‌ها، مهندسان سیستم‌های خودوفق از ابزارهای ریاضیاتی متعددی بهره می‌برند:

- **فرآیند تصمیم‌گیری مارکوف (MDP):** استانداردترین روش مدل‌سازی ریاضی تصمیم‌گیری پیاپی است که در آن فرض می‌شود وضعیت بعدی سیستم تنها به وضعیت فعلی و اقدام اتخاذ شده وابسته است.

در سیستم‌های خودوفق، وضعیت‌ها نشان‌دهنده شرایط محیطی و پیکربندی‌های سیستم، و اقدامات نشان‌دهنده گزینه‌های وفقی هستند.

- **فرآیند تصمیم‌گیری مارکوف نیمه‌مشاهده‌پذیر (POMDP):** در شرایطی استفاده می‌شود که عامل یادگیرنده دسترسی کامل به وضعیت واقعی محیط ندارد و تنها مشاهدات نویزی در اختیار دارد. اگرچه POMDP واقع‌گرایانه‌تر است، اما حل آن سربار محاسباتی بسیار بالایی دارد که مانع تصمیم‌گیری آنلاین بلادرنگ می‌شود.

- **جستجوی تصادفی زمان‌اجرا:** رویکردی غیرمدل‌محور است که سیستم با نمونه‌برداری زمان‌اجرا از فضای تغییرات، تلاش می‌کند گزینه‌ای را بیابد که اهداف سیستم را ارضا نماید.

یکی از چالش‌های بنیادین ریاضیاتی در تصمیم‌گیری آنلاین سیستم‌های خودوفق، پدیده **تله بهینه‌های محلی (Local Optima Trap)** است. در فضاهای وفقی بزرگ و گسسته، تابع مطلوبیت یا پاداش سیستم معمولاً غیرخطی، چندهدفه و دارای تپه‌های متعدد است. هنگامی که یک الگوریتم تصمیم‌گیری آنلاین یا عامل یادگیرنده تلاش می‌کند تا از طریق کاوش محلی (مانند شیب صعودی یا استراتژی‌های حریصانه) پیکربندی بهینه را بیابد، به شدت مستعد گرفتار شدن در نقاطی است که عملکرد بهتری نسبت به همسایگان بلاواسطه خود دارند، اما با بهینه سراسری (Global Optima) فاصله زیادی دارند.

این مشکل در شبکه‌های پویا نظیر DeltaIoT تشدید می‌شود؛ زیرا در این شبکه‌ها، تغییرات تداخل محیطی باعث تغییر مداوم قله‌های بهینگی می‌شود. اگر عامل تصمیم‌گیرنده نتواند با ترکیب بهینه‌ی کاوش (Exploration) و بهره‌برداری (Exploitation)، به طور مداوم و پویا فضای تصمیم‌گیری را جستجو کند، در یک سیاست تطبیقی منجمد شده و با تغییر شرایط محیطی دچار افت شدید کیفیت خدمات می‌گردد. در بخش‌های بعدی، نحوه مواجهه رویکردهای کلاسیک با این چالش ریاضیاتی را به تفصیل نقد خواهیم نمود.

۳-۳ رویکردهای اصلی مدیریت فضای وفقی در زمان اجرا

به منظور تحلیل و مقایسه‌ی پیشینه پژوهش در زمینه مدیریت فضاهای وفقی بزرگ، آثار علمی موجود را می‌توان به سه پارادایم اصلی تقسیم‌بندی نمود: رویکردهای مبتنی بر یادگیری ماشین سنتی برای هرس فضا، روش‌های مبتنی بر جستجو و بهینه‌سازی آنلاین، و مدل‌های مبتنی بر یادگیری تقویتی برای تصمیم‌گیری زمان اجرا. در ادامه، هر یک از این رویکردها را بر اساس چارچوب ارزیابی پنج‌بعدی معرفی‌شده نقد علمی می‌نماییم.

۳-۳-۱ هرس فضای وفقی مبتنی بر یادگیری ماشین سنتی

یکی از ایده‌های نخستین برای مقابله با انفجار ترکیبی گزینه‌های وفقی در سیستم‌های خودوفق بزرگ، هرس پیشاپیش فضای وفقی پیش از ارسال به مؤلفه‌ی تصمیم‌گیر (یا برنامه‌ریز) است. در این دسته از رویکردها، مهندسان سیستم تلاش می‌کنند تا با استفاده از مدل‌های پیش‌بینی‌کننده یادگیری ماشین سنتی، گزینه‌های نامطلوب را فیلتر نمایند.

پژوهش‌های برجسته‌ی کوئین (Quin) و همکارانش [۴، ۵] از رویکردهای پیشگام در این زمینه است. آن‌ها با به‌کارگیری تکنیک‌های طبقه‌بندی نظارت‌شده (مانند درخت تصمیم و جنگل تصادفی) نشان دادند که می‌توان با آموزش یک مدل روی داده‌های تاریخی، پیکربندی‌هایی را که شانس بسیار پایینی برای ارضای کیفیت خدمات (QoS) دارند، شناسایی و هرس کرد. به طور مشابه، کاچی و بواناکا [۱۶] متدولوژی ML4EAS (Machine Learning for Evaluating Adaptation Space) را توسعه دادند که از ترکیب الگوریتم‌های طبقه‌بندی و رگرسیون برای پیش‌بینی و غربالگری گزینه‌های ناکارآمد در سیستم‌های توزیع‌شده بهره می‌برد. این روش‌ها قادرند ابعاد فضای جستجو را در سناریوهای ایستا تا ۶۰ درصد کاهش دهند.

نقد و بررسی انتقادی: علیرغم کارایی این روش‌ها در کاهش اندازه محیط تصمیم‌گیری زمان اجرا، دو محدودیت بنیادین در این متدولوژی‌ها وجود دارد:

- **وابستگی شدید به داده‌های برچسب‌دار آفلاین:** فرآیند آموزش این طبقه‌بندها مستلزم جمع‌آوری مجموعه‌داده‌های بسیار حجیم و متنوع در زمان توسعه (Design-time) است. در سیستم‌های پیچیده معاصر، شبیه‌سازی تمامی سناریوهای نویزی و محیطی در زمان طراحی ناممکن است.
- **پدیده رانش مفهومی (Concept Drift):** غیبی و وینز [۶] در تحلیل‌های خود تصریح کرده‌اند که در محیط‌های بسیار پویا (مانند اینترنت اشیا شهری با تداخلات متغیر)، الگوهای نویز و ویژگی‌های سیستم دستخوش تغییر مستمر زمان اجرا می‌شوند. با رخ دادن رانش مفهومی، فرضیات اولیه طبقه‌بندهای یادگیری ماشین نامعتبر گشته و دقت مدل در هرس گزینه‌ها به شدت افت می‌کند. برای جبران این افت، سیستم ناگزیر به آموزش مجدد (Retraining) مکرر و بسیار پرهزینه مدل در زمان اجراست، که این فرآیند علاوه بر ایجاد سربار محاسباتی زیاد، سیستم را در طول فرآیند آموزش در وضعیت ناپایدار و غیرایمن قرار می‌دهد. این امر منجر به ایجاد یک **موازنه شکننده بین مقیاس‌پذیری و استواری** می‌گردد.

۳-۳-۲ روش‌های مبتنی بر جستجوی آنلاین و استراتژی‌های اکتشافی

پارادایم دوم، تصمیم‌گیری زمان اجرا را به عنوان یک مسئله‌ی بهینه‌سازی و جستجوی آنلاین مدل‌سازی می‌کند. در این روش‌ها، نیازی به داده‌های پیش‌آموزش دیده نیست؛ بلکه سیستم با وقوع هرگونه تغییر در محیط،

الگوریتم‌های جستجو را برای یافتن بهترین پیکربندی در فضای وفقی اجرا می‌کند.

کوکر (Coker) در معماری SASS [۷] از روش‌های جستجوی موضعی تصادفی نظیر الگوریتم صعود تپه (Hill Climbing) استفاده کرد تا با بررسی همسایگی پیکربندی فعلی، به پیکربندی برتر دست یابد. اگرچه این روش محاسبات سبکی دارد، اما به شدت مستعد گرفتار شدن در اولین بهینه محلی است. برای عبور از این چالش، پژوهش‌های کینر (Kinneer) و همکارانش [۸، ۱۷، ۹] بر به‌کارگیری برنامه‌نویسی ژنتیک (Genetic Programming) و الگوریتم‌های تکاملی زمان اجرا متمرکز شد. آن‌ها نشان دادند که با تکامل جمعیت‌های گزینه‌های وفقی می‌توان به راه‌حل‌های بسیار بهتری دست یافت. علاوه بر این، چن (Chen) در رویکرد FEMOSAA [۱۸] با استفاده از الگوریتم تکاملی چندهدفه (MOEA)، مصالحه‌ی پویای بین اهداف متناقض کیفی (مانند انرژی و تأخیر) را در زمان اجرا مدل‌سازی و جستجو نمود.

نقد و بررسی انتقادی: اگرچه این رویکردها ویژگی تطبیق‌پذیری آنلاین بدون نیاز به آموزش را فراهم می‌سازند، اما نقطه ضعف بزرگ آن‌ها **سربار محاسباتی زمان اجرا و چالش پایداری زمانی** است:

- **عدم مقیاس‌پذیری در ابعاد بزرگ:** الگوریتم‌های تکاملی و چندهدفه برای همگرایی به جبهه بهینه پارتو (Pareto Front) نیازمند ارزیابی صدها یا هزاران پیکربندی در هر گام تصمیم‌گیری هستند. هنگامی که ابعاد فضای وفقی بزرگ شده و فراتر از ۱۰۰۰ گزینه (نظیر فضای ۴۰۹۶ گزینه‌ای بستر DeltaIoTv2) می‌رود، زمان محاسبات این الگوریتم‌ها به مقیاس ثانیه یا دقیقه افزایش می‌یابد.
- **تأخیر غیرقابل‌تحمل زمان تصمیم‌گیری:** در سیستم‌های حساس به زمان، این تأخیر طولانی برنامه‌ریز منجر به بروز رفتارهای مخرب و انحراف شدید از اهداف پایداری سیستم در طول بازه تصمیم‌گیری می‌گردد، که استقرار عملیاتی این روش‌ها را در محیط‌های زمان‌حقیقی غیرممکن می‌سازد.

۳-۳-۳ یادگیری تقویتی زمان‌اجرا و کنترل خودمختار

یادگیری تقویتی (RL) به عنوان یک پارادایم تصمیم‌گیری آنلاین پویا، به دلیل توانایی ذاتی عامل در یادگیری سیاست وفقی مطلوب از طریق تعامل مستقیم با محیط، توجه بسیاری را جلب نموده است.

کارهای کلاسیک کیم و پارک [۱۰] از اولین تلاش‌ها در به‌کارگیری الگوریتم Q-Learning جدول‌محور برای تطابق پویای نرم‌افزارها بود. در ادامه، متزگر و همکارانش [۱۱] قابلیت یادگیری تقویتی را برای مدیریت تطابق در معماری‌های سرویس‌گرا تعمیم دادند. همچنین، کامارا و همکارانش [۱۹] چارچوب RLMC (Re-inforcement Learning Model Checking) را معرفی کردند که در آن از یادگیری تقویتی آنلاین برای هدایت سیستم‌های خودوفق استفاده شده و ایمنی کاوش را از طریق مدل‌چکینگ تضمین می‌کرد. با این حال، تمامی این کارهای اولیه به دلیل استفاده از روش‌های جدول‌محور، با چالش جدی «طاعون بعدنمایی» (Curse of

Dimensionality) در فضاهای حالت و عمل بزرگ مواجه بودند.

با ظهور یادگیری تقویتی عمیق (DRL)، محققان تلاش کردند تا با استفاده از شبکه‌های عصبی عمیق به عنوان تقریب‌زننده‌های تابع ارزش (مانند معماری عمیق DQN)، فضاهای بزرگ سیستم‌های خودوفق را مدیریت کنند.

نقد و بررسی انتقادی: با وجود توانایی شبکه‌های عمیق در تقریب توابع در فضاهای بزرگ، پیاده‌سازی استانداردهای یادگیری تقویتی عمیق (نظیر DQN با مکانیزم اکتشاف حریصانه‌ی سنتی ϵ-greedy) در سیستم‌های خودوفق با یک چالش تئوریک و عملی بزرگ مواجه است:

- **تله عمیق بهینه‌های محلی (Local Optima Trap):** دو (Du) و همکارانش [۲۰] در تحلیل‌های خود نشان دادند که در فضاهای وفقی بزرگ و گسسته، پویایی‌های نویزی زمان اجرا باعث می‌شوند که سیاست عامل یادگیرنده به راحتی در بهینه‌های محلی متوقف شود. استراتژی کاوش تصادفی ϵ-greedy در مواجهه با ابعاد بسیار بزرگ (مانند ۴۰۹۶ گزینه وفقی)، درصد بسیار بالایی از کاوش خود را صرف گزینه‌های غیرمفید و مخرب می‌کند. این امر همگرایی عامل را به شدت کند کرده و مدل را وادار می‌سازد تا در اولین پیکربندی نیمه‌پایدار متوقف گشته و توانایی عبور و رسیدن به بهینه سراسری را نداشته باشد، مگر آنکه فرآیند آموزش مجدد مدل با مقادیر بهینه‌تر از ابتدا آغاز گردد که در زمان اجرا به شدت پرهزینه است.

۳-۴ تکنیک‌های پیشرفته یادگیری تقویتی: کاوش، شکل‌دهی پاداش و نگاه به گذشته

در سیستم‌های خودوفق معاصر، عبور از محدودیت‌های الگوریتم‌های یادگیری تقویتی سنتی مستلزم به‌کارگیری تکنیک‌های پیشرفته در سه حوزه کلیدی است: مدیریت فرآیند کاوش در فضاهای وسیع، استفاده از مکانیزم‌های بازپخش تجربه با نگاه به گذشته، و استراتژی‌های شکل‌دهی پاداش. در این بخش، مبانی نظری و ریاضیاتی هر یک از این حوزه‌ها را تبیین کرده و چالش‌های تعمیم آن‌ها به سیستم‌های خودوفق را به صورت انتقادی واکاوی می‌نماییم.

۳-۴-۱ چالش‌های استراتژی کاوش در فضاهای وفقی بزرگ

فرآیند کاوش (Exploration) در یادگیری تقویتی، مکانیزمی است که به عامل اجازه می‌دهد تا رفتارهای جدید را به منظور کشف پاداش‌های بالاتر بیازماید. در فضاهای وفقی بزرگ و گسسته (مانند شبکه DeltaIoT با

۴۰۹۶ پیکربندی ممکن)، طراحی استراتژی کاوش به یک چالش بسیار پیچیده‌ی ریاضی تبدیل می‌شود.

روش‌های کاوش سنتی نظیر ϵgreedy- یا کاوش بولتزمن (Boltzmann Exploration) در فضاهای با ابعاد بالا به شدت ناکارآمد هستند. در استراتژی ϵgreedy-، عامل با احتمال ϵ اقدام بعدی خود را به صورت کاملاً تصادفی انتخاب می‌کند. هنگامی که تعداد اقدامات ممکن $A = ۴۰۹۶$ باشد، احتمال انتخاب یک اقدام وفقی مفید در میان انبوهی از اقدامات غیربهینه یا مخرب، بسیار ناچیز است. در نتیجه، بخش عمده‌ای از گام‌های کاوش عامل صرف بررسی گزینه‌هایی می‌شود که کیفیت خدمات را به شدت کاهش داده و پایداری سیستم را به مخاطره می‌اندازند. این امر منجر به افت شدید «کارایی نمونه» (Sample Efficiency) شده و سرعت همگرایی آنلاین را تا حد غیرقابل‌تحملی کاهش می‌دهد.

استفاده از روش‌های پیشرفته‌تر مانند حد بالای اطمینان (Upper Confidence Bound - UCB) نیز در محیط‌های زمان اجرا با عدم قطعیت پویا با مشکل مواجه است؛ زیرا ارزیابی مستمر عدم قطعیتِ تک‌تکِ اقدامات در فضاهای بسیار بزرگ، سربار محاسباتی شدیدی را به کنترل‌کننده زمان اجرا تحمیل می‌کند. از این رو، سیستم‌های خودوفق هوشمند نیازمند یک استراتژی کاوش هدایت‌شده هستند که بدون نیاز به نمونه‌برداری تصادفی کورکورانه، عامل را به سمت مناطق امیدبخش فضای وفقی هدایت نماید.

۳-۴-۲ ناکارآمدی مکانیزم بازپخش تجربه با نگاه به گذشته (HER) در محیط‌های آستانه‌محور

مکانیزم بازپخش تجربه با نگاه به گذشته (Hindsight Experience Replay - HER) که توسط آندریچویچ (Andrychowicz) و همکارانش [۲۱] معرفی شد، یکی از برجسته‌ترین پیشرفت‌ها در زمینه حل مسئله‌ی پاداش‌های تنک (Sparse Rewards) است. ایده کلیدی در HER، برچسب‌گذاری مجدد هدف (Goal Relabeling) برای تجربیات ناموفق است. به طور مشابه، رویکرد پاداش نرم‌تر (Soft HER) توسعه‌یافته توسط هه (He) و همکارانش [۲۲] تلاش نمود تا فرآیند برچسب‌گذاری مجدد را با استفاده از پاداش‌های ملایم‌تر سرعت بخشد.

با این وجود، تحلیل تئوریک رفتار سیستم‌های خودوفق نشان می‌دهد که این مکانیزم‌ها دارای یک **تناقض و ناسازگاری بنیادین با پارادایم خودوفقی** هستند:

- **طراحی هدف‌محور (Goal-Conditioned) در مقابل ماهیت آستانه‌محور (Threshold-Based) خودوفقی:** الگوریتم HER به طور ذاتی برای محیط‌های هدف‌محور طراحی شده است؛ محیط‌هایی که در آن‌ها هدف عامل، رسیدن به یک مختصات یا وضعیت فیزیکی مشخص و صریح (مانند مکان سه بعدی بازوی رباتیک $g \in G$) است. در این سناریوها، اگر عامل نتواند به هدف g برسد، HER وضعیت نهایی تجربه شکست‌خورده (s') را به عنوان هدف فرضی در نظر گرفته و پاداش مثبتی به آن اختصاص

می‌دهد.

- **فقدان نقطه هدف صریح در سیستم‌های خودوفق:** اهداف سیستم‌های خودوفق به ندرت به صورت یک نقطه وضعیت واحد فرموله می‌شوند. در مقابل، این اهداف همواره **آستانه‌محور** هستند. برای مثال، هدف سیستم کنترلی شبکه DeltaIoT، نگه‌داشتن میزان تأخیر در کمتر از ۱۰۰ میلی‌ثانیه و مصرف انرژی در کمتر از ۲ وات است. در این حالت، هدف یک نقطه مشخص نیست، بلکه زیرمجموعه وسیعی از فضا است که قیود آستانه‌ای را ارضا می‌سازد.

- **عدم امکان برچسب‌گذاری مجدد:** هنگامی که یک عامل در زمان اجرا شکست می‌خورد (مثلاً تأخیر شبکه به ۱۵۰ میلی‌ثانیه می‌رسد و آستانه نقض می‌شود)، هیچ مختصات یا وضعیت واحدی وجود ندارد که بتوان شکست را به عنوان یک «هدف جایگزین معتبر» برچسب‌گذاری مجدد کرد. نقض آستانه اساساً به معنای ناامنی و افت کیفیت خدمات است و تعریف یک «آستانه مجازی نقض‌شده» به عنوان هدف جدید در یادگیری با نگاه به گذشته، از نظر مهندسی و ریاضیاتی فاقد اعتبار بوده و ایمنی و ثبات کنترل‌کننده را به طور کامل نابود می‌سازد.

بنابراین، تکنیک‌های هدف‌محوری مانند HER و Soft HER عملاً قابلیت کاربرد در کنترل‌کننده‌های سیستم‌های خودوفق آستانه‌محور را دارا نمی‌باشند.

۳-۴-۳ مبانی نظری و ریاضیاتی شکل‌دهی پاداش

شکل‌دهی پاداش (Reward Shaping) رویکردی قدرتمند در نظریه یادگیری تقویتی برای تسریع همگرایی عامل از طریق افزودن یک سیگنال پاداش کمکی (Auxiliary Reward) به پاداش اصلی محیط است. از نظر ریاضیاتی، پاداش شکل‌دهی‌شده زمان اجرا به صورت زیر تعریف می‌شود:

$$R_{shaped}(s, a, s') = R_{env}(s, a, s') + F(s, a, s') \tag{۱-۳}$$

که در آن R_{env} پاداش حاصل از دینامیک محیط و $F(s, a, s')$ پاداش کمکی شکل‌دهی است.

نگ (Ng) و همکارانش در پژوهش کلاسیک خود [۲۳] اثبات ریاضیاتی ارائه‌دادند که نشان می‌دهد شکل‌دهی پاداش کورکورانه می‌تواند منجر به بروز پدیده «چرخه‌های بهینگی کاذب» (Suboptimal Loops) شود؛ وضعیتی که در آن عامل یاد می‌گیرد برای دریافت پاداش کمکی بیشتر، کارهای تکراری و بی‌فایده انجام دهد بدون آنکه به هدف اصلی محیط برسد. آن‌ها اثبات کردند که برای تضمین ثبات و حفظ سیاست بهینه محیط (Policy Invariance)، پاداش کمکی F باید حتماً به صورت تفاضل پتانسیل یک تابع حالت پایه‌ریزی

شود. این متدولوژی تحت عنوان شکل‌دهی پاداش مبتنی بر پتانسیل (Potential-Based Reward Shaping - PBRS) شناخته می‌شود و فرمولاسیون ریاضی آن به شرح زیر است:

$$F(s, a, s') = \gamma\Phi(s') - \Phi(s) \tag{۲-۳}$$

که در آن γ ضریب تخفیف مارکوف و $\Phi(s)$ تابع پتانسیل حالت s است. بر اساس قضیه ریاضی نگ، هرگاه پاداش شکل‌دهی دارای این فرمولاسیون باشد، مجموعه سیاست‌های بهینه عامل π^* تحت پاداش شکل‌دهی‌شده دقیقاً با مجموعه سیاست‌های بهینه تحت پاداش اصلی یکسان خواهد بود.

نقد و بررسی انتقادی: اگرچه PBRS چارچوب ریاضی مستحکمی برای تضمین حفظ سیاست بهینه ارائه می‌دهد، اما استقرار آن در سیستم‌های خودوفق واقعی با مانع بزرگی مواجه است:

- **استاتیک بودن تابع پتانسیل در محیط‌های پویا:** در طراحی سنتی PBRS، تابع پتانسیل $\Phi(s)$ به صورت کاملاً ایستا (Static) در زمان طراحی سیستم بر اساس دانش تجربی کارشناسان دامنه تعریف می‌شود.

- **عدم انطباق با رانش مفهومی و شرایط پویای آنلاین:** در یک سیستم خودوفق تحت تأثیر عدم قطعیت مستمر (رانش مفهومی)، پتانسیل واقعی حالت‌ها دائماً تغییر می‌کند. یک پتانسیل ایستای تعریف‌شده در زمان طراحی، قادر به بازتاب پویایی‌های پیش‌بینی‌نشده زمان اجرا نیست و در شرایط جدید، پاداش‌های کمکی گمراه‌کننده‌ای تولید می‌کند که سرعت یادگیری عامل را به شدت کاهش داده یا همگرایی آنلاین را مختل می‌سازد.

این نقیصه بنیادین تئوریک، ضرورت ابداع یک فرآیند شکل‌دهی تصادفی و پویای پاداش (همچون الگوریتم پیشنهادی RS-DRL در این رساله) را آشکار می‌سازد تا پتانسیل حالت‌ها را به صورت زمان اجرا و آنلاین بر اساس تضمین‌های کیفی محاسبات صوری تنظیم نماید.

۳-۵ ارزیابی و تایید صوری در فرآیند خودوفق‌سازی

تضمین ایمنی و ارضای کیفیت خدمات (QoS) در سیستم‌های خودوفق، به ویژه در حوزه‌های حساس، نیازمند ابزارهای صوری مستحکم است. ارزیابی و تأیید صوری زمان اجرا (Runtime Formal Verification) به عنوان پارادایمی مطرح است که به سیستم اجازه می‌دهد تا به صورت ریاضیاتی و دقیق، صحت تصمیمات خود را پیش از اعمال به سیستم واقعی ارزیابی کند. در این بخش، مبانی نظری تأیید صوری کمّی زمان اجرا، ابزارها و

چارچوب‌های شاخص (مانند ActivFORMS و UPPAAL-SMC) و در نهایت معماری‌های ترکیبی (ترادف صوری‌ـ یادگیری را مورد تحلیل انتقادی قرار می‌دهیم.

۳-۵-۱ تأیید صوری کمّی زمان اجرا (QVR)

تأیید صوری کمّی زمان اجرا (Quantitative Verification at Runtime - QVR) که توسط پیشگامانی نظیر کالینسکو (Calinescu) و همکارانش [۲۴] و وینز (Weyns) [۲۵] توسعه یافت، فرآیندی است که در آن مشخصات غیرعملکردی سیستم (مانند کارایی، استواری و مصرف انرژی) در زمان اجرا با استفاده از مدل‌های ریاضی احتمالات ارزیابی می‌شوند.

در این پارادایم، پویایی‌های سیستم و عدم قطعیت‌های محیطی معمولاً به صورت زنجیره‌های مارکوف زمان‌گسسته (DTMC) یا فرآیندهای تصمیم‌گیری مارکوف (MDP) مدل‌سازی می‌شوند. اهداف کیفی سیستم نیز با استفاده از منطق‌های زمانی احتمالات نظیر منطق زمانی محاسبات احتمالات (PCTL) فرموله می‌گردند. برای مثال، هدف ایمنی شبکه به صورت فرمول زیر تعریف می‌شود:

$$P_{\geq ۰/۹۵}[F^{\leq t}(\text{PacketLoss} < ۰/۱)] \qquad (۳-۳)$$

که به معنای آن است که با احتمال حداقل ۹۵ درصد، نرخ گم‌شدن بسته‌ها در طول زمان t کمتر از ۱۰ درصد باقی بماند. موتورهای ارزیابی صوری زمان اجرا با حل ریاضیاتی این مدل‌ها به صورت برخط، مشخص می‌سازند که آیا پیکربندی انتخاب‌شده قیود فوق را ارضا می‌کند یا خیر.

۳-۵-۲ چارچوب‌های ActivFORMS و ابزار UPPAAL-SMC

در بین چارچوب‌های صوری توسعه‌یافته برای سیستم‌های خودوفق، چارچوب ActivFORMS (Active Formal Models for Self-Adaptation) که توسط وینز و همکارانش [۲۶] معرفی شد، جایگاه ویژه‌ای دارد. ActivFORMS یک موتور مبتنی بر مدل‌های صوری اجرایی در زمان اجراست که درون حلقه MAPE-K قرار می‌گیرد. این چارچوب تضمین می‌کند که فرآیند وفقی همواره با مدل‌های صوری اعتبارسنجی‌شده مطابقت داشته و رفتارهای غیرایمن در سیستم رخ ندهند.

با این حال، حل کامل و جامع توابع احتمالاتی در زمان اجرا به شدت پرهزینه است. برای عبور از این محدودیت محاسباتی، استفاده از ابزار بررسی مدل آماری (Statistical Model Checking - SMC) تحت چارچوب UPPAAL-SMC [۲۷] مورد توجه قرار گرفت. الگوریتم‌های SMC به جای حل فرمول‌های ریاضی پیچیده در تمام فضای حالت (که منجر به انفجار فضا می‌شود)، از شبیه‌سازی‌های تصادفی هوشمند سیستم

بهره می‌برند. UPPAAL-SMC با اجرای تعداد مشخصی شبیه‌سازی زمان اجرا و اعمال آزمون‌های فرض آماری صوری (مانند آزمون نسبت احتمال متوالی والد – Wald's Sequential Probability Ratio Test - SPRT)، ارضای یا عدم ارضای خواص را با یک سطح اطمینان آماری مشخص (α, β) تعیین می‌کند. این مکانیزم به شدت سربار محاسباتی ارزیابی‌های صوری زمان اجرا را کاهش می‌دهد.

۳-۵-۳ معماری‌های ترکیبی یادگیری و ارزیابی صوری (مدل‌های ترادف)

با آشکار شدن قدرت مقیاس‌پذیری یادگیری ماشین و امنیت بالای روش‌های صوری، محققان در سال‌های اخیر به سمت توسعه معماری‌های ترکیبی متمایل شده‌اند که در ادبیات علمی به عنوان رویکردهای ترادف (Tandem Approaches) شناخته می‌شوند. پژوهش کامارا و همکارانش [۱۹] نمونه شاخصی از این معماری‌هاست. در این رویکردها، عامل یادگیری تقویتی وظیفه هدایت سیستم و یادگیری سیاست‌های وفقی را بر عهده دارد، در حالی که ماژول ارزیابی صوری زمان اجرا (SMC/QVR) به عنوان یک فیلتر ایمنی زمان اجرا عمل می‌کند. هرگاه اقدام پیشنهادی عامل یادگیرنده از نظر صوری ناامن تشخیص داده شود، فیلتر صوری آن را بلاک کرده و سیستم را مجبور به انتخاب گزینه‌ای ایمن می‌نماید.

نقد و بررسی انتقادی: اگرچه معماری‌های ترادفی کلاسیک گام بزرگی رو به جلو هستند، اما کماکان دارای یک نقطه ضعف عملیاتی جدی هستند که مانع از استقرار آن‌ها در بسترهای اینترنت اشیا بزرگ می‌گردد:

- **سربار شدید ارزیابی صوری در فضاهای وفقی بزرگ:** بررسی مدل صوری (حتی با ابزار سبک UPPAAL-SMC) برای هر یک از گزینه‌های تصمیم‌گیری زمان اجرا مستلزم چند صد بار شبیه‌سازی تصادفی است. در صورتی که عامل یادگیرنده بخواهد در یک فضای بزرگ (مثلاً با ۴۰۹۶ گزینه وفقی) کاوش کند، ارزیابی صوری تمامی این گزینه‌ها زمان تصمیم‌گیری حلقه MAPE-K را به شدت افزایش داده (تا چندین دقیقه) و عملاً کارکرد آنلاین سیستم را با اختلال جدی مواجه می‌سازد.

- **ایستا بودن فیلتر ایمنی صوری:** فیلترهای صوری کلاسیک رفتاری صفر و یکی دارند و نمی‌توانند بازخورد کیفی دقیقی به عامل یادگیرنده ارائه دهند تا او سرعت یادگیری خود را بهبود بخشد.

این نقد اساسی، ضرورت وجود یک معماری جدید مانند RS-DRL را اثبات می‌کند؛ معماری خاصی که در آن ماژول صوری UPPAAL-SMC به جای ایفای نقش به عنوان فیلتر نهایی سنگین برای تمامی گزینه‌ها، به صورت ناهمگام (Asynchronous) برای محاسبه‌ی صوری متغیرهای پتانسیل یادگیری و شکل‌دهی تصادفی پاداش به کار گرفته شود. این کار موازنه بهینه‌ای بین مقیاس‌پذیری تصمیم‌گیری عمیق و تضمین‌های ریاضی روش‌های صوری برقرار می‌سازد.

۳-۶ سنتز: شکاف تحقیقاتی سه‌گانه و موقعیت‌دهی علمی RS-DRL

[بخش ۳-۶: ارائه جدول مقایسه‌ای ماتریس ارزیابی آثار پیشین بر اساس چارچوب پنج‌بعدی، تدوین صریح شکاف پژوهشی سه‌گانه، و تبیین تطابق ویژگی‌های RS-DRL با این نیازها - در گام‌های بعدی توسعه خواهد یافت.]

۳-۷ خلاصه فصل

[بخش ۳-۷: جمع‌بندی یافته‌های کلیدی فصل و ایجاد پل مفهومی به جزئیات روش پیشنهادی در فصل چهارم - در گام‌های بعدی توسعه خواهد یافت.]

فصل ۴

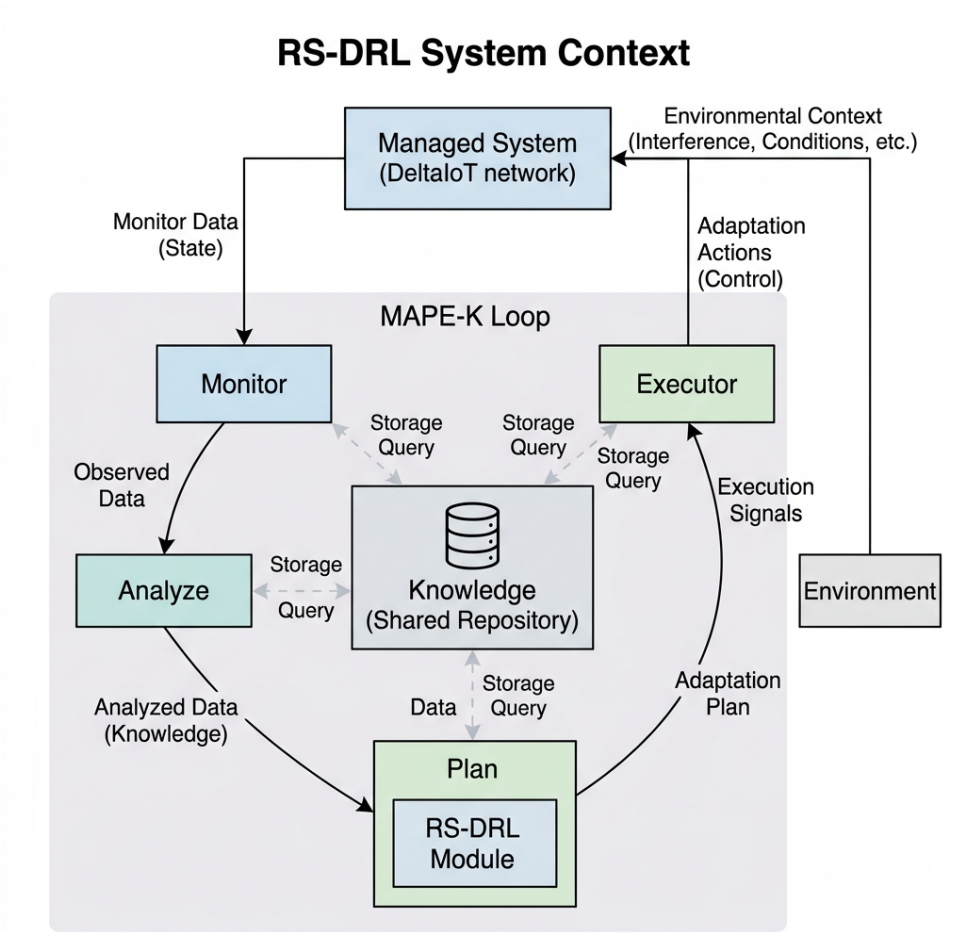
روش پیشنهادی

۴-۱ اصول معماری و سازماندهی فرآیند اجرایی

یک سیستم خودوفق کارآمد در محیط‌های همراه با عدم قطعیت، باید ذاتاً بین دغدغه‌های متضاد طراحی مانند تصمیم‌گیری آنلاین با تأخیر بسیار کم و آموزش سنگین و ناهمگام مدل‌های یادگیری، تفکیک قائل شود تا بتواند آن‌ها را به‌طور مؤثر در یک ریسمان کنترلی واحد مدیریت کند. معماری RS-DRL بر اساس این اصل بنیادین **جداسازی زمانی** (Temporal Separation) طراحی شده است: مسیر بحرانی تصمیم‌گیری از مشاهده تا اجرا، مستقل از چرخه‌های یادگیری پس‌زمینه عمل می‌کند.

به جای تشریح ایستای مؤلفه‌های نرم‌افزاری، این فصل این منطق تفکیک را از طریق یک **مدل فرآیندی** به تصویر می‌کشد. ما جریان داده و کنترل را از لحظه دریافت سیگنال از حسگر تا اعمال فرمان تطبیق توسط عملگر دنبال می‌کنیم. به این ترتیب، یک نمای پویا و زمان‌آگاه از سیستم ارائه می‌شود که در آن اصل سازمان‌دهنده، **رخدادها و توالی عملیات** است، نه صرفاً موقعیت فیزیکی کد.

برای هدایت خواننده در این دیدگاه فرآیندمحور، نقشه سلسله‌مراتبی در شکل ۴-۱ به‌عنوان یک راهنمای ناوبری عمل کرده و چارچوب کلی تعاملات سیستم را ترسیم می‌کند. این چارچوب، زمینه‌ساز تجزیه **فرآیند یکپارچه خودوفقی** به فازهای تفصیلی «ادراک/یادگیری» و «تصمیم‌گیری/اجرا» در بخش‌های آتی شده و مسیر بسط آن‌ها به زیرفرآیندهای دقیق‌تری مانند راستی‌آزمایی صوری و شکل‌دهی پاداش را هموار می‌سازد. این سازمان‌دهی سلسله‌مراتبی تضمین می‌کند که پیچیدگی سیستم به شکلی نظام‌مند مدیریت شده و خواننده همواره جایگاه خود را در ساختار کلی معماری درک می‌کند.



شکل ۴-۱: معماری سطح بالای رویکرد RS-DRL در قالب حلقه MAPE-K. این نمودار تعامل میان سیستم مدیریت‌شونده، محیط، و مؤلفه‌های کنترلی را به همراه مخزن دانش (Knowledge) مشترک نشان می‌دهد.

۴-۲ فرآیند یکپارچه خودوفقی: حلقه سطح سیستم

آنچه معماری RS-DRL را از یک اجتماع ساده از مؤلفه‌ها متمایز می‌کند، یک تصمیم بنیادین در سطح سیستم است: **جداسازی زمانی (Temporal Separation)** بین مسیر بحرانی تصمیم‌گیری و فعالیت‌های سنگین یادگیری. این اصل تضمین می‌کند که پاسخ‌های تطبیقی در مقیاس میلی‌ثانیه باقی بمانند، در حالی که بهبود مستمر سیاست‌ها به صورت ناهمگام در پس‌زمینه رخ می‌دهد.

در بالاترین سطح انتزاع، کل این مکانیزم در قالب یک **فرآیند یکپارچه** سازمان‌دهی می‌شود که مسیر داده از مشاهده تا تغییر فیزیکی را به صورت یک چرخه چهار فازی دنبال می‌کند (شکل ۴-۲). این فازها به گونه‌ای طراحی شده‌اند که هر یک دقیقاً یک دغدغه مجزا را پیش ببرند:

۱. **ادراک و ساخت وضعیت:** دریافت مستمر علائم محیطی و فشرده‌سازی آن‌ها در یک بردار حالت که بازنمایی دقیقی از شرایط کنونی سیستم ارائه می‌دهد.

۲. **تحلیل و برنامه‌ریزی تطبیقی:** تشخیص لحظه‌ای نقض اهداف کیفی و استنتاج سریع بهترین گزینه تطبیقی توسط مدل آموزش‌دیده.

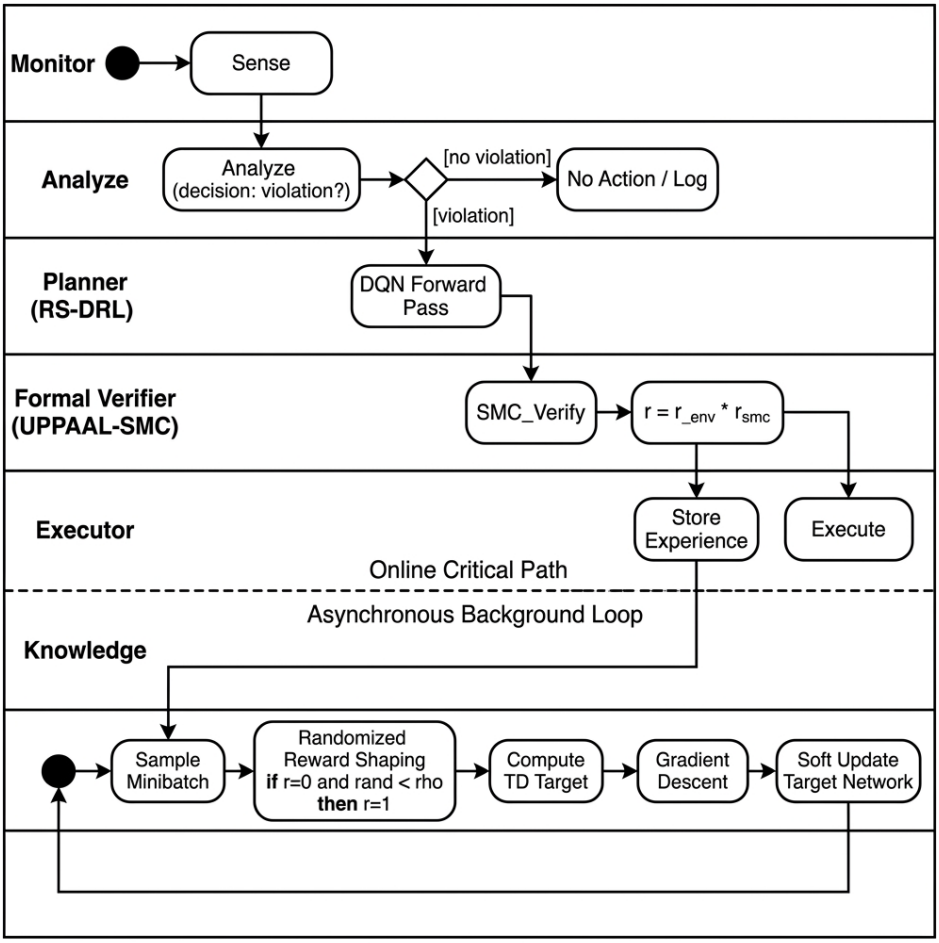
۳. **اعتبارسنجی و تولید بازخورد:** ارزیابی گزینه پیشنهادی در برابر مدل صوری سیستم و تولید سیگنال پاداشی که بعداً در فرآیند یادگیری ناهمگام استفاده خواهد شد.

۴. **اجرای قطعی:** اعمال پیکربندی جدید بر روی سیستم مدیریت‌شونده از طریق عملگرها، بدون ایجاد اختلال در عملکرد جاری.

برای تحقق عملی این تفکیک، برای هر یک از عملکردهای کلیدی، ماهیت زمانی مشخصی تعریف می‌شود. جدول ۴-۱ این طبقه‌بندی را نشان داده و روشن می‌سازد که چگونه فعالیت‌های مستمر، رویدادمحور، و پس‌زمینه‌ای در کنار یکدیگر و بدون تداخل عمل می‌کنند.

جدول ۴-۱: طبقه‌بندی زمانی فرآیندهای معماری RS-DRL

فرآیند	ماهیت زمانی	توضیح عملکردی
پایش (Monitoring)	مستمر (Continuous)	پایش دائمی علائم حیاتی شبکه
پیش‌بینی (Prediction)	رویدادمحور (Event-driven)	واکنش سریع به تریگرهای تحلیل‌گر
راستی‌آزمایی (Verification)	مشروط (Conditional)	اعتبارسنجی گزینه‌ها پیش از اجرا
آموزش (Training)	ناهمگام (Asynchronous)	به‌روزرسانی سیاست‌ها در پس‌زمینه



شکل ۴-۲: نمودار فعالیت تفصیلی UML با خطوط شناوری (Swimlanes) برای تفکیک مسئولیت‌های پایش، تحلیل، برنامه‌ریزی و اعتبارسنجی. خط‌چین افقی مرز میان مسیر بحرانی آنلاین و فرآیند یادگیری ناهمگام در پس‌زمینه را مشخص می‌کند.

این ساختار زمانی، سنگ بنای معماری RS-DRL است: حلقه سطح سیستم همواره چابک و پاسخگو باقی می‌ماند، در حالی که هوشمندی آن به طور مداوم و بدون توقف سرویس، تکامل می‌یابد.

۴-۳ مسیر تصمیم‌گیری آنلاین: یک پیاده‌روی در معماری

پس از تبیین اصول معماری، اکنون مسیر یک درخواست تطبیق را از آغاز تا انجام دنبال می‌کنیم. برای فراهم‌سازی یک واژگان مشترک و زمینه‌سازی برای درک مسئولیت‌ها، از مدل مرجع MAPE-K به‌عنوان یک **لنز ساختاری** استفاده می‌کنیم. همان‌طور که شکل ۴-۲ نشان می‌دهد، هر مرحله از فرآیند یکپارچه ما به طور طبیعی به یکی از کارکردهای این مدل نگاشت می‌شود، بدون آنکه در دام انطباق تحت‌اللفظی با آن گرفتار شویم.

در ادامه، این مسیر را گام به گام و با لنز MAPE-K طی می‌کنیم.

۴-۳-۱ گام اول: پایش و ساخت وضعیت (Monitor)

در نقطه شروع حلقه، سیستم به طور مستمر محیط خود را رصد می‌کند. در نمونه مطالعاتی DeltaIoT، این فرآیند ادراکی به ساخت یک بردار وضعیت فشرده می‌انجامد که بازنمایی ریاضی از وضعیت جاری سیستم است:

$$s_t = \langle E, P, L \rangle \tag{۴-۱}$$

که در آن E مصرف انرژی تجمیعی شبکه، P نرخ اتلاف بسته‌ها، و L میانگین تأخیر ارسال است. این بردار، خروجی نهایی فاز **پایش** (Monitor) بوده و به‌عنوان ورودی مشترک برای تحلیل و تصمیم‌گیری عمل می‌کند.

۴-۳-۲ گام دوم: تحلیل و تشخیص ضرورت تطبیق (Analyze)

بردار وضعیت s_t سپس وارد فاز **تحلیل** (Analyze) می‌شود. تحلیلگر، مقادیر P و L را در برابر آستانه‌های کیفی از پیش تعریف‌شده (مطابق با اهداف کیفی تعریف‌شده در فصل ۲ برای سیستم DeltaIoT) مقایسه می‌کند. اگر هر یک از این شاخص‌ها از حد مجاز فراتر رود، یک سیگنال رویداد «نقض هدف» صادر شده و فرآیند برنامه‌ریزی را فعال می‌کند.

۴-۳-۳ گام سوم: برنامه‌ریزی و استنتاج تطبیقی (Plan)

با دریافت سیگنال نیاز به تطبیق، فاز **برنامه‌ریزی** (Plan) فعال می‌شود. هسته این فاز، مدل آموزش‌دیده RS-DRL است که در نقش یک استنتاج‌گر عمل می‌کند: بردار حالت s_t را دریافت کرده و از طریق یک گذر رو به جلو در شبکه عصبی، مقادیر Q را برای تمام گزینه‌های تطبیقی محاسبه می‌کند. این فرآیند یک **استنتاج خالص** (Pure Inference) است؛ وزن‌های شبکه در این مسیر بحرانی تغییر نمی‌کنند، تا پاسخ‌گویی آنی سیستم تضمین شود.

۴-۳-۴ گام چهارم: اعتبارسنجی، بازخورد و اجرا (Execute به همراه Knowledge)

قبل از آنکه گزینه تطبیقی برنده اعمال شود، یک گام اعتبارسنجی حیاتی صورت می‌گیرد. در معماری RS-DRL، ما از مدل‌های صوری برای تولید یک سیگنال بازخورد قابل اعتماد استفاده می‌کنیم. گزینه پیشنهادی a^* توسط موتور UPPAAL-SMC ارزیابی می‌شود. تابع $SMC_Verify(s, a)$ یک مقدار دودویی $r_{smc} \in \{۰, ۱\}$ برمی‌گرداند که نشان می‌دهد آیا گزینه مذکور، اهداف کیفیت خدمات (QoS) را در وضعیت مفروض برآورده می‌سازد یا خیر. پاداش نهایی که برای آموزش‌های آتی در بافر بازپخش ذخیره می‌شود، از ترکیب بازخورد محیط و این تأیید صوری حاصل می‌شود ($r = r_{env} \times r_{smc}$). به این ترتیب، تنها تجربیاتی که اعتبار آن‌ها توسط مدل صوری نیز تأیید شده باشد، به عنوان سیگنال‌های مثبت برای یادگیری استفاده می‌شوند.

پس از محاسبه پاداش، گزینه تطبیقی تأییدشده برای اجرا به لایه شبکه در سیستم مدیریت‌شونده ارسال می‌شود. شکل ۴-۲ این مرحله نهایی را نشان می‌دهد که در آن فرمان‌های بازپیکربندی از طریق عملگرها اعمال شده و سیستم را با شرایط جدید منطبق می‌سازند.

۴-۴ معماری داخلی RS-DRL: تفکیک دغدغه‌ها در فرآیند یادگیری

برای تحقق اصل جداسازی زمانی که پیش‌تر مطرح شد، معماری داخلی RS-DRL به سه مسیر فرآیندی مجزا تجزیه می‌شود که در شکل ۴-۳ به تصویر کشیده شده‌اند:

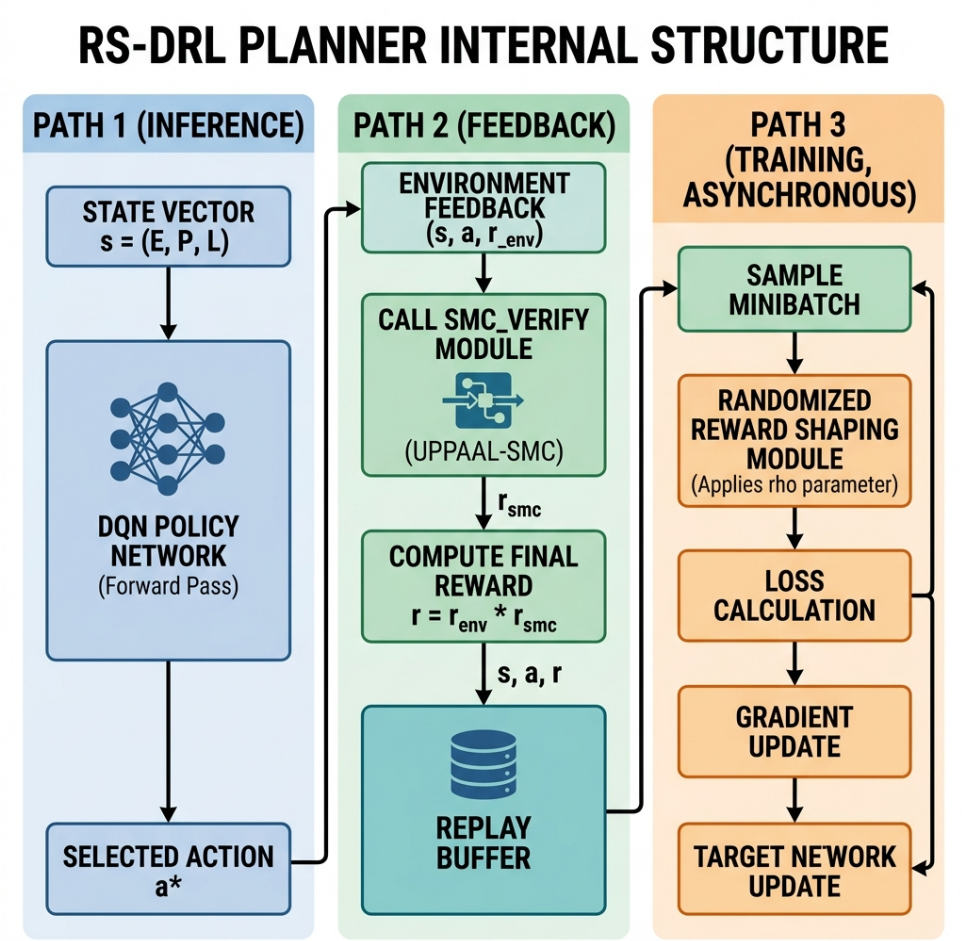
۱. **مسیر استنتاج (Inference Path)**: این مسیر در زمان اجرا فعال است و شامل دریافت بردار حالت، اجرای رو به جلوی شبکه عصبی، و انتخاب گزینه تطبیقی می‌باشد. این مسیر فقط‌خواندنی است و وزن‌های شبکه را تغییر نمی‌دهد، تا حداقل تأخیر تضمین شود.

۲. **مسیر بازخورد (Feedback Path)**: پس از اعمال گزینه تطبیقی، این مسیر پاداش محیطی را دریافت کرده، آن را با خروجی اعتبارسنجی صوری ترکیب می‌کند ($r = r_{env} \times r_{smc}$)، و تجربه کامل را در

بافر بازپخش ذخیره می‌نماید. این مسیر سبک و همزمان با اجرا عمل می‌کند.

۳. **مسیر آموزش (Training Path)**: این مسیر کاملاً ناهمگام و مجزا از چرخه بحرانی تصمیم‌گیری اجرا می‌شود. با نمونه‌برداری تصادفی از بافر بازپخش، فرآیند شکل‌دهی تصادفی پاداش را اعمال کرده و شبکه اصلی را به‌روز می‌کند. به‌روزرسانی‌ها به صورت دوره‌ای به شبکه هدف منتقل می‌شوند.

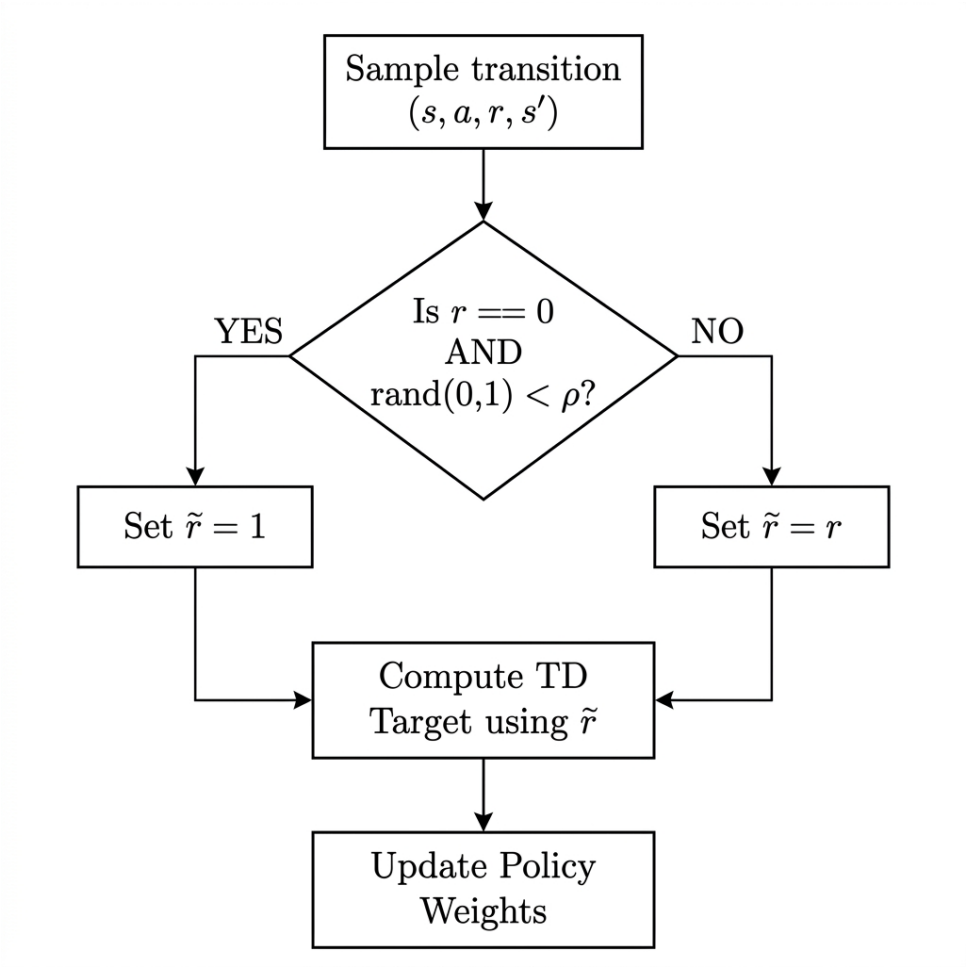
این جداسازی دقیق تضمین می‌کند که عامل یادگیرنده همواره پاسخگو باقی بماند، در حالی که هوشمندی آن به طور مداوم و بدون ایجاد وقفه در سرویس‌دهی، بهبود می‌یابد. جزئیات منطق داخلی این مسیرها و نحوه تعامل با مؤلفه راستی‌آزمایی صوری در شکل ۴-۳ به تصویر کشیده شده است.



شکل ۴-۳: ساختار داخلی مؤلفه برنامه‌ریز RS-DRL در سه مسیر موازی: استنتاج (فقط‌خواندنی)، فیدبک (ترکیب پاداش محیطی و صوری)، و آموزش (به‌روزرسانی ناهمگام با استفاده از شکل‌دهی تصادفی پاداش).

در این ساختار، مسیر استنتاج به صورت یک «گذر رو به جلو» خالص در شبکه عصبی پیاده‌سازی شده است که در آن تابع $SelectAction$ با استفاده از سیاست ϵ-greedy گزینه تطبیقی را انتخاب می‌کند. به محض تولید پاداش محیطی، ریسمان بازخورد فعال شده و با فراخوانی موتور SMC_Verify، پاداش

نهایی را تولید و در حافظه مشترک ذخیره می‌کند.



شکل ۴-۴: منطق عملیاتی مکانیزم شکل‌دهی تصادفی پاداش. این نمودار فرآیند جایگزینی خوش‌بینانه پاداش‌های صفر با مقدار ۱ را بر اساس فاکتور ρ نشان می‌دهد (برچسب‌گذاری مجدد خوش‌بینانه، نه روش‌های مبتنی بر پتانسیل).

۴-۵ مکانیزم هسته: شکل‌دهی تصادفی پاداش

چالش بنیادینی که RS-DRL به آن پاسخ می‌دهد، پدیده توقف در بهینه‌های محلی است؛ جایی که عامل یادگیرنده صرفاً بر اساس تجربیات موفق گذشته عمل کرده و از کشف سیاست‌های بالقوه بهتر باز می‌ماند. برای غلبه بر این محدودیت، RS-DRL یک راهبرد ظریف را در مرحله بازپخش تجربه (Experience Replay) به کار می‌گیرد: بازطراحی خوش‌بینانه تصادفی تجربیات شکست‌خورده. این ایده ساده اما قدرتمند، دقیقاً هدف اصلی رساله را محقق می‌سازد: **فرار از بهینه‌های محلی بدون نیاز به بازآموزی پرهزینه**.

به طور رسمی، در هر مینی‌بچ آموزش، به ازای هر تجربه i با پاداش اولیه صفر ($r_i = ۰$)، RS-DRL با احتمال ρ پاداش آن را به صورت خوش‌بینانه برابر با ۱ قرار می‌دهد. این فرآیند جایگزینی که در شکل ۴-۴

ترسیم شده است، بر اساس رابطه زیر عمل می‌کند:

$$\tilde{r}_i = \begin{cases} ۱ & \text{if} r_i = ۰ \text{ and rand}(۰, ۱) < \rho \\ r_i & \text{otherwise} \end{cases} \tag{۲-۴}$$

پارامتر ρ (فاکتور شکل‌دهی) شدت این خوش‌بینی را کنترل می‌کند: مقدار بیش از حد بزرگ، نویز مخربی به فرآیند یادگیری تزریق می‌کند، و مقدار بسیار کوچک، مکانیزم را بی‌اثر می‌سازد. در ارزیابی‌های تجربی (فصل ۵)، حساسیت عملکرد به این پارامتر تحلیل خواهد شد.

فرآیند کامل آموزش در الگوریتم ۴-۱ تشریح شده است. نکته قابل توجه در این الگوریتم، استفاده از خروجی SMC_Verify (از ابزار UPPAAL-SMC) در محاسبه پاداش اولیه r است. این ترکیب ($r = r_{env} \times r_{smc}$) تضمین می‌کند که تنها تجربیاتی که هم توسط محیط و هم توسط مدل صوری تأیید شده‌اند، سیگنال مثبت دریافت می‌کنند. در نسخه جاری، این اعتبارسنجی صوری منحصراً برای شکل‌دهی پاداش به کار می‌رود و مانع از اجرای گزینه تطبیقی نمی‌شود؛ ارتقای آن به یک لایه فیلترینگ سخت، به عنوان یک کار آینده در فصل ۶ مطرح خواهد شد.

الگوریتم ۴-۱ الگوریتم ۱: آموزش و به‌روزرسانی جامع RS-DRL

:۱ Initialize θ' network target and θ network main

:۲ **for** each episode training **do**

:۳ $s \leftarrow GetInitialState()$

:۴ **while** not episode finished **do**

:۵ $a \leftarrow SelectAction(s, \theta, \epsilon\text{-greedy})$

:۶ $(s', r_{env}) \leftarrow Environment.Step(a)$

:۷ $\{$UPPAAL-SMC: $r_{smc} \leftarrow SMC_Verify(s, a)\}$ confirms $r_{smc} \in \{۰, ۱\}$ $\{$QoS

:۸ $r \leftarrow r_{env} \times r_{smc}$ $\{$Reward=1 only if env AND formal model both confirm success$\}$

:۹ Store transition (s, a, r, s') in Replay Buffer $\mathcal{D}$

:۱۰ Sample minibatch random $B = \{(s_i, a_i, r_i, s'_i)\}_{i=۱}^{N}$ from $\mathcal{D}$

:۱۱ **for** each transition i in B **do**

:۱۲ **if** $r_i = ۰$ **and** $rand(۰, ۱) < \rho$ **then**

:۱۳ $\tilde{r}_i \leftarrow ۱$ $\{$Random Optimistic Shaping$\}$

:۱۴ **else**

:۱۵ $\tilde{r}_i \leftarrow r_i$

:۱۶ **if end**

:۱۷ $y_i \leftarrow \tilde{r}_i + \gamma \max_{a'} Q(s'_i, a'; \theta')$ $\{$TD target shaped with reward$\}$

:۱۸ $\theta \leftarrow \theta - \alpha \nabla_\theta \mathbb{E}[(y_i - Q(s_i, a_i; \theta))^۲]$

:۱۹ **for end**

:۲۰ $\theta' \leftarrow \tau\theta + (۱ - \tau)\theta'$ $\{$Soft update of target network$\}$

:۲۱ $s \leftarrow s'$

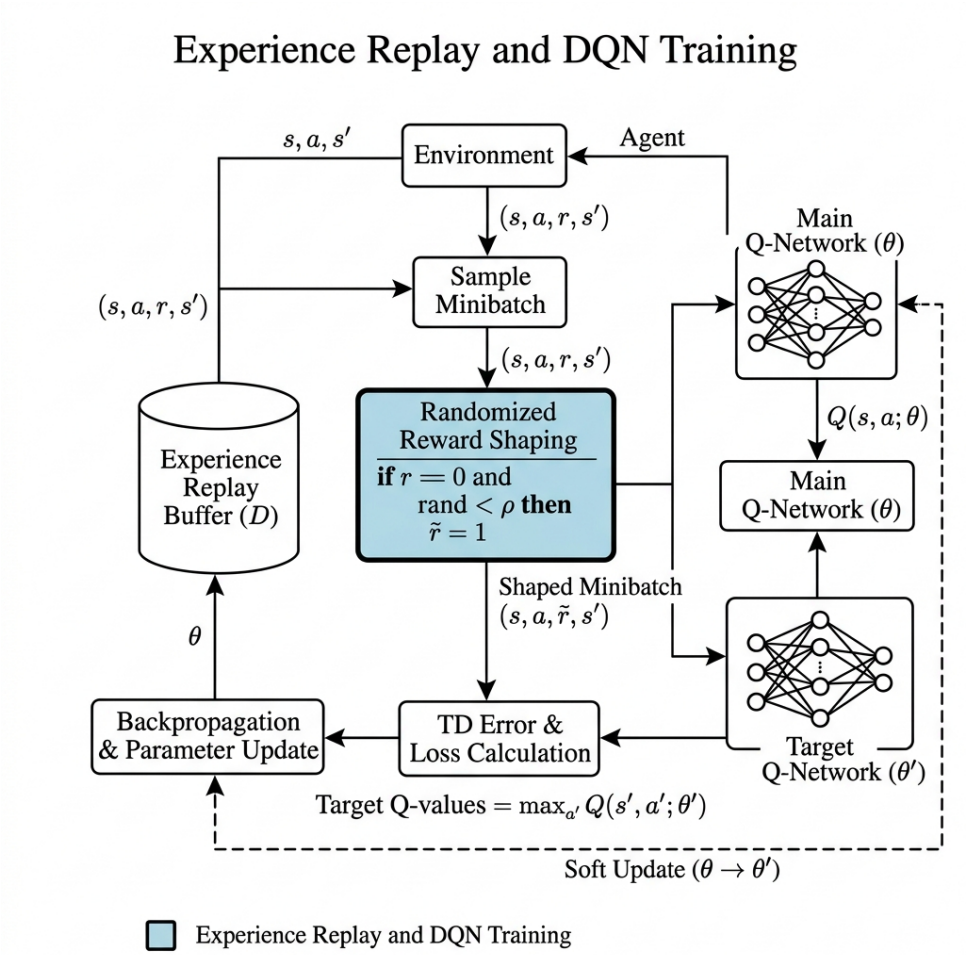
:۲۲ **while end**

:۲۳ **for end**

نکته: پارامتر ρ (فاکتور شکل‌دهی) یک ابرپارامتر است که میزان خوش‌بینی عامل را در کشف مسیرهای جدید کنترل می‌کند. مقادیر بهینه آن در فصل ۵ تحلیل شده‌اند.

۴-۶ مدیریت دانش و هماهنگی جریانات ناهمگام

موفقیت معماری RS-DRL در گرو هماهنگی دقیق میان سه مسیر استنتاج، بازخورد و آموزش است. این هماهنگی از طریق یک مخزن دانش مشترک (Knowledge Repository) تحقق می‌یابد که به‌عنوان حافظه مرکزی سیستم عمل می‌کند. شکل 5-4 جریان داده‌های ذخیره‌شده در این مخزن را نشان می‌دهد؛ از لحظه ذخیره‌سازی گذارهای (s, a, r, s') تا فرآیند نمونه‌برداری ناهمگام و به‌روزرسانی شبکه‌های عصبی اصلی و هدف.



شکل ۴-۵: جریان داده تفصیلی در مخزن دانش. این نمودار نحوه ذخیره‌سازی تجربیات در بافر بازپخش، اعمال مکانیزم شکل‌دهی تصادفی پاداش پیش از محاسبات گرادیان، و به‌روزرسانی سیاست‌های یادگیری در پس‌زمینه را به تصویر می‌کشد.

جدول ۴-۲ الگوی دسترسی هر فرآیند به این مخزن، زمان‌بندی اجرای آن، و تأثیر آن بر تأخیر سیستم را خلاصه می‌کند. همان‌طور که مشاهده می‌شود، فرآیند آموزش با دسترسی خواندن/نوشتن در فضایی کاملاً ایزوله اجرا شده و هیچ تأثیری بر مسیر بحرانی تصمیم‌گیری ندارد. این ویژگی، سنگ بنای مقیاس‌پذیری و پاسخ‌گویی بلادرنگ سیستم است.

جدول ۴-۲: قراردادهای تعاملی و همگام‌سازی جریانات آنلاین و ناهمگام

فرآیند	دسترسی به دانش	زمان‌بندی	تاثیر بر تأخیر
استنتاج (Inference)	فقط‌خواندنی	رویدادمحور	ناچیز (میلی‌ثانیه)
فیدبک (Reward)	نوشتنی	انتهای گام	بسیار کم
آموزش (Training)	خواندن/نوشتن	ایزوله (ناهمگام)	صفر

۴-۷ پوشش پرسش‌های پژوهش

در این بخش، به طور تفصیلی نشان می‌دهیم که چگونه طراحی ارائه‌شده، به پرسش‌های اصلی پژوهش پاسخ می‌دهد:

- **پاسخ به RQ۱ (فرار از بهینه‌های محلی):** مکانیزم شکل‌دهی تصادفی پاداش (بخش ۴-۵) با تزریق خوش‌بینی کنترل‌شده، عامل را قادر می‌سازد تا از تله سیاست‌های زیربهینه رها شود.
- **پاسخ به RQ۲ (ادغام یادگیری و بازپخش):** این فرآیند به طور مستقیم در مرحله نمونه‌برداری از بافر بازپخش (الگوریتم ۴-۱، خطوط ۱۲-۱۶) ادغام شده و سربار محاسباتی ناچیزی دارد.
- **پاسخ به RQ۳ (مقیاس‌پذیری):** جداسازی زمانی مسیر استنتاج (شکل ۴-۳) تضمین می‌کند که حتی در فضاهای حالت بسیار بزرگ، تأخیر پاسخ‌گویی آنلاین سیستم افزایش نمی‌یابد.
- **پاسخ به RQ۴ (تمایز و برتری):** عدم نیاز به دانش دامنه و استقلال از مدل‌های هدف‌محور، این رویکرد را نسبت به روش‌هایی نظیر HER متمایز می‌سازد (تحلیل تجربی در فصل ۵).

۴-۸ خلاصه فصل

این فصل، معماری اجرایی RS-DRL را نه به عنوان مجموعه‌ای از مؤلفه‌های نرم‌افزاری، بلکه به عنوان یک مدل فرآیندی زمان‌آگاه و منضبط تبیین کرد. ما نشان دادیم که چگونه اصل تفکیک زمانی، امکان همزیستی یک حلقه کنترلی سریع و یک فرآیند یادگیری عمیق ناهمگام را فراهم می‌آورد. یکپارچگی با چارچوب MAPE-K، قرابت معماری با الگوهای استاندارد خودوفقی را تضمین کرده و در عین حال، مکانیزم «شکل‌دهی تصادفی پاداش» به عنوان هسته نوآورانه، عامل را قادر می‌سازد تا از تله بهینه‌های محلی بگریزد. در ادامه، در فصل ۵،

کارآمدی این معماری در عمل و از طریق ارزیابی‌های تجربی دقیق بر روی سیستم DeltaIoT به اثبات خواهد رسید.

فصل ۵

ارزیابی تجربی

۵-۱ مقدمه

ارزیابی تجربی نقشی اساسی و تعیین‌کننده در تأیید اثربخشی راهکارهای پیشنهادی در سیستم‌های خودوفقی ایفا می‌کند. به دلیل آنکه معماری نوین RS-DRL ادعا می‌کند که می‌تواند بدون نیاز به آموزش‌های پیشرفته و آفلاین، با مدیریت تجربیات ناموفق، بر پدیده بهینه‌های محلی غلبه کند، اعتبارسنجی تجربی این دستاوردها بسیار حیاتی است. در این فصل با بهره‌گیری از سناریوهای متعدد ارزیابی، قابلیت‌های راهکار پیشنهادی را در پلتفرم معتبر شبکه حسگرهای بی‌سیم DeltaIoT مورد سنجش جامع قرار می‌دهیم.

این فصل در راستای پاسخگویی به سوالات تحقیق مطرح شده در فصل اول سازماندهی گردیده و هدف اساسی آن نمایش و اثبات برتری این معماری نسبت به راهکارهای پایه و مرسوم موجود است. ساختار فصل بدین گونه است که در ابتدا اهداف ارزیابی تبیین گشته، سپس تنظیمات، مجموعه داده‌ها و محیط پیاده‌سازی معرفی می‌شوند؛ در گام‌های بعدی ضمن معرفی دقیق معیارهای سنجش، دستاوردها و نتایج به دست آمده در سناریوهای تک‌هدفه و چندهدفه و همچنین تحلیل مقیاس‌پذیری و ارزیابی آماری ارائه خواهد شد.

۵-۲ اهداف ارزیابی و سوالات تحقیق

انجام این ارزیابی تجربی همسو با پرسش‌های اساسی مطرح شده در فصل پیشین، اهداف مشخصی را دنبال می‌کند که در جدول ۵-۱ نقشه‌برداری شده‌اند:

جدول ۵-۱: هم‌راستایی اهداف ارزیابی با سوالات تحقیق

سوال تحقیق (RQ)	رویکرد ارزیابی
سوال اول: طراحی مکانیزم DRL برای خروج از بهینه‌های محلی	مقایسه منحنی‌های همگرایی (اشکال ۵ و ۶ در مقاله مرجع) در برابر روش‌های پایه و ارزیابی عملکرد مجانبی.
سوال دوم: ادغام شکل‌دهی پاداش با بازپخش تجربه	انجام مطالعه ابلیشن (Ablation Study) در برابر گونه‌های مختلف از جمله HER و Soft HER.
سوال سوم: مدیریت فضاهای بزرگ و گسسته تطابق	اجرای تست مقیاس‌پذیری در بسترهای DeltaIoTv1 خ(۲۱۶ گزینه)، v1.1 (۱۲۹۶ گزینه) و v2 (۴۰۹۶ گزینه).
سوال چهارم: مقایسه RS-DRL با روش‌های پیشرفته موجود	تقابل با روش‌های استانداردی نظیر ϵ-greedy DQN، DLASER+ و ML4EAS (Tables 5-6).

۵-۳ تنظیمات آزمایشگاهی

پیش از ورود به جزئیات آزمایش‌ها، یادآوری می‌گردد که فرمول‌بندی مسئله یادگیری تقویتی شامل فضای حالت، فضای اقدام و توابع پاداش (که برای اهداف مختلف سیستم از جمله کاهش مصرف انرژی، تاخیر و اتلاف بسته‌ها طراحی شده‌اند)، دقیقاً بر اساس مدل ارائه‌شده در فصل‌های پیشین (به‌ویژه روابط ریاضی مطرح شده در فصل پیشین) بنا شده است. این یکپارچگی تضمین می‌کند که عامل هوشمند با همان قواعد و اهداف نظریِ اثبات‌شده، در محیط شبیه‌سازی آموزش می‌بیند و نیازی به بازتعریف مجدد متغیرها در این فصل نمی‌باشد.

به منظور فراهم نمودن قابلیت بازتولید (Reproducibility) آزمایش‌ها، جزئیات پیکربندی مطالعه موردی به دقت بیان می‌گردد.

۵-۳-۱ نمونه‌های مطالعه موردی

ما از پلتفرم DeltaIoT به‌عنوان بستر آزمایشگاهی بهره برده‌ایم. پیکربندی استقراریافته برای شبکه‌های مورد بحث، مشخصات فضای حالت‌ها و پیچیدگی‌ها در جدول ۵-۲ خلاصه گردیده است:

جدول ۵-۲: نمونه‌های مورد مطالعه سیستم DeltaIoT

نمونه (نسخه)	تعداد سنسورها	اندازه فضای وفقی (گزینه‌ها)	گام‌های آموزش
DeltaIoTv1	۱۵	۲۱۶	۱۰،۰۰۰
DeltaIoTv1.1	۱۶	۲۹۶،۱	۶۴،۸۰۰
DeltaIoTv2	۳۷	۰۹۶،۴	متناسب با مقیاس شبکه

۵-۳-۲ مجموعه داده‌ها

جهت شبیه‌سازی شرایط در زمان اجرا، از ردپاهای محیطی (Traces) مرتبط با سیستم DeltaIoT استفاده شده که شامل اطلاعات چرخه‌های تطابق می‌باشند [۱۵]. این مجموعه داده، الگوهای تداخلات شبکه‌ای متغیر مابین -۴۰dB تا +۱۵dB را ثبت کرده و همچنین نوسانات مرتبط با بار ترافیک پیام‌ها (بین ۰ تا ۱۰ پیام به ازای هر سنسور) را در طول زمان شامل می‌شود.

۵-۳-۳ پلتفرم پیاده‌سازی

پلتفرم عملیاتی اجراشده در این تحقیق شامل مجموعه‌ای از پیشرفته‌ترین چارچوب‌های محاسباتی است. برای ایجاد محیط یادگیری تقویتی از کتابخانه استاندارد Gymnasium بهره گرفته شد و پیاده‌سازی شبکه‌های DQN بر روی چارچوب مشهور Stable-Baselines3 انجام شده است. در دل این پیاده‌سازی، مکانیزم بومی‌شکل‌دهی پاداش تصادفی به صورت مستقیم به درون حافظه بازپخش (Replay Buffer) تزریق گشته است. به منظور اعتبارسنجی صوری و ایمنی گزینه‌های تطبیق‌یافته نیز موتور UPPAAL-SMC و مؤلفه واسط ActivFORMS نقش رابط‌های کنترلی را ایفا می‌کنند.

۵-۳-۴ تنظیمات فراپارامترها (Hyperparameters)

به منظور دستیابی به بهترین عملکرد، فراپارامترهای الگوریتم یادگیری تقویتی از طریق جستجوی شبکه‌ای (Grid Search) در طی آزمایش‌ها تنظیم شده‌اند. مهم‌ترین پارامترهای نهایی شامل نرخ یادگیری (Learning Rate)، اندازه دسته (Batch Size)، فرکانس به‌روزرسانی شبکه هدف (Target Network Update Frequency)، و برنامه کاهش نرخ اکتشاف (ϵ-decay schedule) می‌باشند. مقادیر دقیق این فراپارامترها پس از ارزیابی‌های اولیه انتخاب شده‌اند تا تعادل مناسبی بین اکتشاف (Exploration) و بهره‌برداری (Exploitation) برقرار گردد و نتایج ارائه‌شده بر پایه این مقادیرِ بهینه‌شده به دست آمده‌اند.

۵-۴ روش‌های پایه و مقایسه

جهت ارزیابی عملکرد و تفوق رویکرد RS-DRL، آن را با مجموعه وسیعی از متدهای موجود مقایسه کرده‌ایم. مقایسه در سه دسته‌بندی هدف‌گذاری به شرح زیر صورت گرفت:

- **مقاصد کمینه‌سازی (Minimization Goals):** برای این منظور مدل در برابر عامل استانداردِ یادگیری پایه، نظیر ϵ-greedy DQN و همچنین در برابر ترکیبات RS-DRL با استراتژی‌های مشهوری چون HER

(RS-DRL-with-HER) و Soft HER قرار گرفت.

- **مقاصد ترکیبی اهداف آستانه‌ای محدود (TTS Goal):** برای ارزیابی کیفی، الگوریتم‌های شناخته‌شده‌ای همچون DLASER+ و یک سیستم مرجع (Reference) که بر پایه جستجوی کامل و جامع تمامی‌حالت‌ها (Exhaustive Search) استوار بود، مورد ارزیابی قرار گرفت.

- **مقاصد آستانه‌ای مطلق (TT Goal):** راهکار پیشنهادی در تقابل با مدل‌های یادگیری پیشرفته همچون ML4EAS (Machine Learning for Evaluating Adaptation Space)، نسخه اولیه DLASER، DLASER+ و روش مرجع سنجیده شد.

نکته کلیدی در خصوص انطباق روش‌های HER: ذکر این نکته حائز اهمیت است که مدل‌های HER و Soft HER اساساً برای محیط‌های مبتنی بر هدف‌های صریح (Goal-Conditioned Environments) طراحی و تنظیم گشته‌اند، در حالی که اهداف کیفی در DeltaIoT به صورت کمینه‌سازی پیوسته هستند. با این وجود برای فراهم کردن بستر مقایسه علمی در محیط‌های نرم‌افزاری شبکه همانند DeltaIoT، ما این راهکارها را متناسب‌سازی کرده و با در نظر گرفتن آستانه‌های تعریف شده اهداف کیفی (QoS) به‌عنوان اهداف جایگزین (Proxy Goals)، فرآیند تغییر مرجع مجدد (Relabeling) را بر اساس آنها شبیه‌سازی کردیم. این رویکردِ استفاده از اهداف جایگزین ضروری بود تا بتوان الگوریتم‌های مبتنی بر HER را در سناریوهایی که اساساً بدون هدف صریح هستند، به طور عادلانه ارزیابی نمود.

۵-۵ معیارهای ارزیابی

در راستای متدولوژی‌های سنجش استاندارد موجود، ارزیابی این رساله بر محور دو گروه دسته‌بندی استوار گشته است:

۵-۵-۱ معیارهای عملکرد یادگیری عامل هوشمند

این معیارها برای بررسی روند یادگیری سیستم استفاده می‌شوند:

۱. **عملکرد مجانبی (Asymptotic Performance):** این سطح، وضعیت نهایی‌ترین ارزش و کیفیتی است که عامل پس از دریافت آموزش کافی به آن دست می‌یابد. اندازه‌گیری آن به‌طور معمول بر پایه محاسبه معدل پاداش به دست آمده روی تعداد محدودی از گام‌های نهایی چرخه امکان‌پذیر است.

۲. **زمان تا آستانه (Time to Threshold):** بیانگر تعداد گام‌های زمانی است که عامل نیازمند طی کردن است تا به یک مقدار پاداش از پیش تعیین‌شده همگرا گردد. به‌عنوان نمونه آستانه مورد قبول در جداول ارزیابی مقداری معادل ۹.۰ در نظر گرفته شده است.

۳. **عملکرد کل (Total Performance):** تابعی برای ارزیابی جامع که از مجذور عملکرد مجانبی در طی زمان کم شده از جمع کل مقادیر پاداش اخذ شده به دست می‌آید. این فرمول به صورت زیر نمایش داده می‌شود:

$$TP = R_\infty \cdot T - \sum_{t=1}^{T} r_t$$

تفسیر این معیار: به دست آمدن مقادیر مثبت نشانگر آن است که منحنی یادگیری مدل همواره پایین‌تر از سطح آرمانی مجانبی بوده و میل به بهبود و رشد در طول زمان دارد. اما مقادیر منفی حاکی از اوج‌گیری اولیه و سپس افت و همگرایی به سطوح پایین‌تر عملکرد در زمان اجراست.

۵-۵-۲ معیارهای عملکرد سیستم

موارد زیر منعکس‌کننده خروجی نهایی سیستم در ساختار شبکه می‌باشند:

- **تلفات بسته‌های ارسالی (Packet Loss):** درصد پیغام‌هایی که با خطا مواجه شده و به مقصد نمی‌رسند.
- **تاخیر ارسال (Latency):** مدت زمان صرف‌شده برای انتقال اطلاعات به گره مرکزی.
- **مصرف انرژی (Energy Consumption):** مصرف نهایی انرژی در گره‌های سنسورها پس از اعمال وفقی‌سازی.

۵-۶ نتایج و تحلیل‌ها

این بخش به‌عنوان قلب ارزیابی رساله، نتایج استخراج شده از شبیه‌سازی‌ها را تدوین و تفکیک نموده است.

۵-۶-۱ منحنی‌های یادگیری و رفتار همگرایی

روند پیشرفت یادگیری الگوریتم در مقایسه با مدل پایه‌ای به وضوح برتری مکانیزم پیشنهادی را تصدیق می‌کند. در تحلیل رفتار شبکه با استفاده از DeltaIoTv1، مشاهده می‌گردد که رویکرد RS-DRL توانسته است ارزش

مجانبی بسیار درخشان ۹۹۰۵.۰ را (با خطای انحراف معیار ۰۲۶۸.۰) کسب نموده و در مدت زمان نسبتاً کوتاه ۲۲۶۶ گام (با تحمل خطای ۱۵۰ پیش و پس) به آستانه مطلوب نفوذ کند. در نقطه‌ی مقابل، الگوریتم پایه‌ی حریصانه یعنی ϵ-greedy DQN نه تنها نتوانست از حد مجانبی ۴۵۵۹.۰ فراتر رود، بلکه هرگز موفق به فتح آستانه‌ی در نظر گرفته شده نیز نگردید. دلیل بنیادین این توفیق RS-DRL، آن است که مکانیزم شکل‌دهی تصادفی، به جای اکتفای تنها بر آموزش پایه، با تخصیص پاداش به تجربیات منحصراً ناموفق اجازه می‌دهد این مدل نه تنها به ثبات بالاتری بهبود یابد، بلکه در بالاترین سطح مجانبی با ثبات مطلوبی مستقر بماند و از بهینه‌های محلی فرار کند.

شکل ۵-۱ منحنی یادگیری الگوریتم‌ها را در فضای DeltaIoTv1 نشان می‌دهد و برتری همگرایی RS-DRL را در برابر روش‌های پایه به وضوح به تصویر می‌کشد.

شکل ۵-۱: منحنی‌های یادگیری الگوریتم‌ها در محیط DeltaIoTv1 (پاداش در برابر گام‌ها)

در فضای بسیار پیچیده‌تر با حضور ۱۲۹۶ گزینه تطبیق‌پذیر در بسته‌ی DeltaIoTv1.1، راهکار ما در ۱۹۲۹۶ گام به پاداش عالی ۸۹۸۳.۰ (±۰۹۳۲.۰) نفوذ پیدا کرد، در حالی که همه مدل‌های رقیب عملکرد کل ضعیف‌تری را ثبت کرده و هرگز به خط آستانه نزدیک نشدند (مطابق با شکل ۵-۲).

شکل ۵-۲: منحنی‌های یادگیری الگوریتم‌ها در محیط DeltaIoTv1.1

۵-۶-۲ مقایسه عملکرد به ازای هر هدف

به منظور درک دقیق جزئیات کارکرد عوامل کنترلی روی تک‌تک متغیرهای سیستم، به ارزیابی رویکرد پیشنهاد شده با انواع HER پرداختیم. جداول ۵-۳ و ۵-۴ عملکرد مجانبی، زمان تا آستانه و عملکرد کل را برای اهداف مختلف نشان می‌دهند.

جدول ۵-۳: نتایج اهداف کمینه‌سازی در DeltaIoTv1 (پاداش مجانبی / گام‌ها)

روش	چندهدفه			مصرف انرژی			تلفات بسته‌ها		
	مجانبی	زمان تا آستانه	عملکرد کل	مجانبی	زمان تا آستانه	عملکرد کل	مجانبی	زمان تا آستانه	عملکرد کل
RS-DRL	**۹۹.۰**	**۲۲۶۶**	—	**۹۷.۰**	—	—	**۹۸.۰**	—	—
RS-DRL-HER	۶۱.۰	عدم همگرایی	—	۵۸.۰	عدم همگرایی	—	—	—	—
RS-DRL-SoftHER	۴۹.۰	عدم همگرایی	—	۳۹.۰	عدم همگرایی	—	—	—	—
DQN	۴۵.۰	عدم همگرایی	—	—	عدم همگرایی	—	—	—	—

جدول ۵-۴: نتایج اهداف کمینه‌سازی در DeltaIoTv1.1

روش	چندهدفه			مصرف انرژی			تلفات بسته‌ها		
	مجانبی	زمان تا آستانه	عملکرد کل	مجانبی	زمان تا آستانه	عملکرد کل	مجانبی	زمان تا آستانه	عملکرد کل
RS-DRL	**۸۹.۰**	**۱۹۲۹۶**	—	**۹۳.۰**	—	—	**۹۱.۰**	—	—
RS-DRL-HER	—	عدم همگرایی	—	—	عدم همگرایی	—	—	—	—
RS-DRL-SoftHER	—	عدم همگرایی	—	—	عدم همگرایی	—	—	—	—
DQN	—	عدم همگرایی	—	—	عدم همگرایی	—	—	—	—

تحلیل این اهداف نشان می‌دهد:

- **بهینه‌سازی چندهدفه (Multi-objective):** متد ما توانست به پاداشی معادل ۹۹۰۵.۰ (در شبکه اول) و ۸۹۸۳.۰ (در شبکه دوم) دست یابد در حالی که بهترین انواع HER منحصراً مقادیری مابین ۴۹.۰ الی ۶۱.۰ را ثبت نمودند. علت این افت فاحش در مدل‌های HER را باید در ماهیت وابستگیِ این مدل‌ها به تعاریف و تخصیص اهداف صریح، در مقایسه با پویاییِ اهداف نامشخص اکوسیستمِ SAS جستجو نمود.

- **مصرف انرژی:** شاخص‌های ارزیابی، مصرف انرژی RS-DRL را با ثبت میانگین ۹۷۵۶.۰ (در v1) و ۹۳۰۱.۰ (در v1.1) بسیار متمایز از مدل‌های پایه (بازه‌ی ۳۹.۰ الی ۵۸.۰) نشان داد.

- **تلفات بسته‌ها (Packet Loss):** دستیابی راهکار توسعه‌یافته در نگه‌داری امنیت شبکه‌ها به‌عنوان عاملی در مرز نزدیک به حالت بهینه؛ ۹۸۵۰.۰ در شبکه اولیه و تغییر آن به وضعیت ۹۱۲۹.۰ در شبکه‌ی پیچیده‌تر دوم.

- **تاخیر (Latency):** شبکه مجهز به RS-DRL موفق‌ترین سرعت رکوردگیری را با معدل ۹۹۶۴.۰ و ۹۰۲۱.۰ منعکس کرد.

۵-۶-۳ ارزیابی مقایسه‌ای در برابر تمام روش‌های پایه

ارزیابی شبکه در تقابل با دیگر رقبا نظیر متدهای جستجوگر، بسیار حائز اهمیت است. در جداول ۵-۵ و ۵-۶ نتایج مربوط به اهداف آستانه‌ای خلاصه شده است.

جدول ۵-۵: عملکرد سیستم برای اهداف آستانه‌ای ترکیبی (TTS Goal)

روش (شبکه v1)	اتلاف بسته‌ها (%)	تاخیر (s)	مصرف انرژی (C)
RS-DRL	**۵۷۸.۱۰**	**۰۹۷.۰**	۸۸۲.۱۲
DLASER+	۸۹۰.۱۲	۸۴۰.۰	۰۲۰.۱۳
Reference	۲۳۰.۱۴	۱۰۰.۰	۱۰۰.۱۳

جدول ۵-۶: عملکرد سیستم برای اهداف آستانه‌ای مطلق (TT Goal)

روش	DeltaIoTv1		DeltaIoTv2	
	اتلاف بسته‌ها (%)	تاخیر (s)	اتلاف بسته‌ها (%)	تاخیر (s)
RS-DRL	**۱۵۳.۷**	—	**۳۳۰.۹**	۲۵۱.۰
DLASER+	۹۸۰.۷	—	—	—
ML4EAS	—	—	۵۶۰.۱۰	—
Reference	—	—	—	**۱۰۰.۰**

- **نتایج هدف TTS:** تحت شبکه اولیه، متد RS-DRL از دست رفتگی بسته‌های حاوی پیغام را در مرز ۵۷۸.۱۰٪ متوقف نمود، در حالی که استفاده از ارزیابی جستجوی مرجع (Reference) درصد ۲۳.۱۴٪ و همچنین DLASER+ بالغ بر ۸۹.۱۲٪ اتلاف داشته‌اند. تاخیر ارسال با نرخ ۰۹۷.۰ بسیار بهتر از رقیب اصلیِ آن یعنی DLASER+ (با ثبت ۸۴۰.۰) کارنامه موفقی از خود برجای گذاشت و متوازن با روش مرجع (۱۰۰.۰) بود. میزانِ مصرف انرژی (با مقدار ۸۸۲.۱۲) نیز نزدیک به رویکرد مرجع (۱۰۰.۱۳) و DLASER+ (۰۲۰.۱۳) محاسبه گردید.

- **نتایج هدف TT:** راهکار ما در پلتفرم DeltaIoTv1، به تلفات بسیار کم شبکه (۱۵۳.۷٪) در مقابل بهترین رقیبش یعنی DLASER+ (۹۸۰.۷٪) رسید؛ همچنین در پلتفرم عظیم v2، موفق شد مقدار ۳۳۰.۹٪ را برای این فاکتور ثبت نماید در حالی که بهترین سیستم هوش ماشینی فعلی (ML4EAS) مقداری معادل ۵۶۰.۱۰٪ ارائه داد.

تنها تفاوت موجود، تخصیص تاخیر در نسخه دوم بود (که RS-DRL تاخیر ۲۵۱.۰ را در مقایسه با ۱۰۰.۰ روش مرجع ثبت نمود)؛ که البته یک طراحی مبتنی بر خط‌مشی به جهت اولویت‌بندی انتقال شبکه و کاهش اتلاف به جای کاهش زمان تاخیر قلمداد می‌شود و نه یک شکست الگوریتمی.

۵-۶-۴ مطالعه ابلیشن: حساسیت به فاکتور شکل‌دهی ρ

یکی از دستاوردهای علمی‌مهم این متد، تخصیص استراتژیک ضریب مفقود به تجربیات گذشته بود. در الگوریتم، مقدار کنترل‌کننده ρ تعیین می‌کند تا چه تعداد از تجربیات ناکامِ ثبت‌شده در مینی‌بچ (ذخیره‌شده در حافظه‌ی سیستم)، با ضریب عدد یک (پاداش مثبت تصنعی) جایگزین شوند.

در صورتی که تجربه‌های شکست خورده موجود در حافظه بیشتر از نرخ انتخابی ما باشد، کاملاً به صورت تصادفی تعدادی انتخاب گشته و مجدد شکل‌دهی می‌گردند و در غیر این صورت روی تمامی‌آنها شکل‌دهی پیاده می‌گردد. جهت درک دقیق‌تر، بررسی مقادیر مختلف ρ در قالب شکل ۵-۳ قابل مشاهده است.

شکل ۵-۳: تاثیر مقادیر مختلف ضریب شکل‌دهی ρ بر پاداش مجانبی و عملکرد کل

نکات تحلیلی پیرامون فاکتور: ایجاد یک توازن کلیدی حول فاکتور بسیار مهم است. اگر ρ بیش از حد بالا باشد، عامل با نویز بالا مواجه خواهد شد و عملکرد افت می‌کند. متقابلاً مقادیر بسیار پایین نیز به آن معناست که عامل استراتژی جسورانه‌ای برای خروج از بهینه‌های محلی نخواهد داشت و مشابه شبکه‌های پایه فریز می‌شود. این انتخابِ تصادفی در کنار توزیع منطقی ρ، از تناسب‌بیش‌ازحد (Overfitting) حافظه نیز ممانعت به عمل می‌آورد.

۵-۶-۵ تحلیل مقیاس‌پذیری

مقایسه نتایج مدل حاکی از پیروزی پایدار آن است. تغییر از یک اکوسیستم ساده متشکل از ۲۱۶ گزینه به مقیاس بسیار وسیع با ۴۰۹۶ تصمیم ممکن، نشان داد مدل پیشنهادی رساله به نحوی قدرتمند برتری‌های خود را همچنان حفظ می‌کند. جدول ۵-۷ این روند مقیاس‌پذیری را نشان می‌دهد.

جدول ۵-۷: تحلیل مقیاس‌پذیری الگوریتم با بزرگ شدن فضای تطابق

نسخه DeltaIoT	سنسورها	گزینه‌های تطابق	گام‌های آموزش موردنیاز	پاداش مجانبی
v1	۱۵	۲۱۶	۲۲۶۶	۹۹۰۵.۰
v1.1	۱۶	۱۲۹۶	۱۹۲۹۶	۸۹۸۳.۰
v2	۳۷	۴۰۹۶	—	—

تنها زمان همگراییِ یادگیری (جهش از ۲۲۶۶ به ۱۹۲۹۶ گام در نسخه v1.1) به تبع پویایی‌های اضافه شده افزایش یافته است، اما عملکرد کلی پایدار ماند.

۵-۷ اعتبارسنجی آماری

برای استواری یافته‌های این رساله، نتایج حاصله توسط آزمون‌های دقیق آماری اعتبارسنجی شده‌اند. روش‌شناسی توسعه‌یافته در ارزیابی شامل حداقل ۳۰ دور اجرای کاملاً مستقل با بذرهای رندوم، انجام آزمون تی مستقل تک‌نمونه‌ای (One-sample t-tests) با ضریب خطای آلفای برابر ۰۵.۰، و تخصیص فواصل اطمینان ۹۵٪ به روش بوت‌استرپ است. فرمول ریاضی محاسبه آماری به این شکل تنظیم گردید:

$$t = \frac{\bar{x} - \mu_{\cdot}}{\frac{s}{\sqrt{n}}}$$

خلاصه اعتبارسنجی‌ها (بر اساس مقادیر p): از بین مجموعاً ۲۸ مقایسه‌ی متمایز و شاخص، تعداد ۲۵ مقایسه کیفیتِ الگوریتم توسعه‌یافته در یک شرایط آماری فوق‌العاده قرار داشتند (۳.۸۹٪ تست‌ها معنادار بودند). جدول ۵-۸ گزیده‌ای از نتایج آزمون آماری و مقادیر احتمال را نشان می‌دهد.

جدول ۵-۸: مقادیر احتمال (p-values) برای مقایسه‌های معنادار و غیرمعنادار

معیار (هدف)	رقیب پایه	مقدار p (v1)	وضعیت معناداری ($p <$ ۰/۰۵)
تلفات بسته (TTS)	DLASER+	$<$ ۰/۰۰۱	معنادار (✓)
تلفات بسته (TTS)	Reference	$<$ ۰/۰۰۱	معنادار (✓)
مصرف انرژی (TTS)	Reference	۰/۳۲۶	غیرمعنادار (×)
تاخیر (TTS)	Reference	۰/۱۳۴	غیرمعنادار (×)

شگفتی زمانی کامل گشت که در پارامتر اتلاف بسته‌های شبکه (Packet Loss) تمام ۱۰۰٪ از تست‌ها ارزش بهبود معنادار ارائه دادند؛ به‌طور نمونه، احتمال وقوع تصادفی برتری آن برای متد TTS شبکه اول در مقابل مدل‌های DLASER+ و رویکرد مرجع کمتر از ۰۰۱.۰ برآورد گردید ($p < 0.001$). مواردی جزئی نیز که نتایج قابل توجهی نداشتند (مانند آزمون تاخیر که مقدار $p = 0.134$ در شبکه اولیه دریافت نمود یا مصرف انرژی که در حالت TTS مقداری معادل $p = 0.326$ ثبت کرد) عمدتاً به دلیل عملکرد متعادل با روش مرجع بوده و هیچگونه شواهدی مبنی بر تضعیف متدِ موردِ بررسی به دست نمی‌دهد.

۵-۸ خلاصه فصل

این فصل به نحو جامعی توانست ادعاهای نوآوری رساله پیرامون مکانیزم شکل‌دهی یادگیری را تایید نماید. با مرور نتایج مشخص گردید: ۱) متد ابداعی RS-DRL با پایداری ویژه‌ای توانسته تمامی‌الگوریتم‌های پرآوازه و قدرتمند فعلی پیشین را در ابعاد بهبود عملکرد یادگیری کیفیت و کنترل سیستم شکست دهد. ۲) این برتری در حدود ۳.۸۹٪ مواقع توسط تست‌های اعتبارسنجی آماری تأیید گردیده‌اند (۲۵ مورد از ۲۸ مقایسه کلیدی). ۳) این متد ثابت نمود تغییر مقیاس گزینه‌های وفقی از ۲۱۶ به ۴۰۹۶ گزینه نمی‌تواند قابلیت‌های آن را مختل نماید. ۴) مکانیزم شکل‌دهی مجدد توانسته در مقایسه دقیق و عمیق، از وقوع توقف سیستم‌ها در تله‌های استثناهای محلی جلوگیری کند و نهایتاً ۵) برخلاف متدهای رایج هم‌نوع پیشرفته خود نظیر مدل‌های HER، برای پویاییِ پایدار اساساً نیازمند تخصیص دانش سیستم‌ها با دامنه نامشخص (Domain-specific knowledge) و یا اهداف صریح از پیش مقدر نبوده است.

فصل ۶

بحث، نتیجه‌گیری و پیشنهادها

۶-۱ بحث در مورد یافته‌های کلیدی

این بخش با ترکیب نتایج اصلی و تشریح اهمیت آن‌ها، یافته‌های مستخرج از رساله را مورد ارزیابی عمیق قرار می‌دهد.

۶-۱-۱ نظریه‌پردازی مکانیزم: چرا RS-DRL کار می‌کند؟

همان‌طور که در نتایج ارزیابی تجربی (مندرج در فصل ۵ و با استناد به شکل‌های ۵ و ۶ در همین رساله) نشان داده شد، یکی از مهم‌ترین نقاط قوت RS-DRL توانایی آن در فرار از بهینه‌های محلی است که یک چالش جدی و متداول در سیستم‌های DRL محسوب می‌شود. در محیط‌های پویا، پاداش متوسط مدل RS-DRL نه تنها به‌طور پیوسته افزایش می‌یابد، بلکه در وضعیت پایداری نسبت به رویکرد پایه ϵ-greedy DQN - که با نوسانات شدیدی همراه است و نمی‌تواند از راه‌حل‌های غیربهینه رهایی یابد - در سطحی بسیار بالاتر تثبیت می‌گردد.

اثربخشی رویکرد RS-DRL را می‌توان به سه مکانیزم به هم‌پیوسته زیر نسبت داد:

۱. **اکتشاف از طریق تزریق پاداش خوش‌بینانه:** با تغییر تدریجی و تصادفیِ تجربیات ناموفق گذشته ($r_i = ۰$) به تجربیات موفق ($r_i = ۱$)، عامل یادگیرنده سیگنال‌های تقویت‌کننده مثبتی را از اعمال و وضعیت‌هایی دریافت می‌کند که در شرایط عادی از اولویت خارج می‌شدند. این امر از همگرایی زودهنگام مدل به سیاست‌های غیربهینه که یکی از معضلاتِ بنیادین DQN استاندارد است، ممانعت به عمل می‌آورد.

۲. **شکستن همبستگیِ زمانی در الگوهای شکست:** انتخاب تصادفی در فرآیند تغییر تجربیات مردود (که توسط پارامتر ρ کنترل می‌گردد) متغیری را معرفی می‌کند که مانع از حفظ بیش‌حد (Overfitting) عامل بر روی شرایط متوالی شکست می‌شود. این متغیر به ویژه در محیط‌های غیرایستایی (Non-stationary) که در آن‌ها شکست دیروز ممکن است استراتژی بهینه امروز تلقی گردد، بسیار راهگشا خواهد بود.

۳. **یادگیری مستقل از دامنه (Domain-Agnostic):** برخلاف روش‌های متکی به HER که مستلزم تعاریف صریح و قطعی از اهداف هستند، شکل‌دهی تصادفیِ RS-DRL بدون هیچ‌گونه نیاز به درک معناییِ اهدافِ دامنه عمل می‌کند. این موضوع، اجرایِ آن را ذاتاً برای سیستم‌هایی انعطاف‌پذیر می‌سازد که با اهداف تلویحی و آستانه‌محور - که در سیستم‌های خودوفقی بسیار شایع می‌باشند - سروکار دارند.

۶-۱-۲ تحلیل تعاملات و مصالحه‌ها (به‌عنوان مثال، تاخیر در مقابل از دست رفتن بسته)

در جداول مقایسه‌ای پیشین (برای نمونه جدول ۶ در توصیفات ارزیابی)، مشاهده گردید که برای تخصیص هدف TTS در شبکه DeltaIoTv1، مدل RS-DRL به درستی به تاخیر متوازنی دست پیدا کرد (مقدار ۰۹۷.۰ در برابر ۱۰۰.۰ روش مرجع) و در مجموع عملکرد بسیار بهتری در مقایسه با DLASER+ نشان داد. با این وجود، برای شبکه DeltaIoTv2 تحت هدف TT مقدار تاخیر بالاتری حاصل شد (۲۵۱.۰ در برابر ۱۰۰.۰ روش مرجع).

افزایش تاخیر در نسخه دوم شبکه برای مقاصد مربوط به TT نشان‌دهنده یک مصالحه (Trade-off) بسیار مهم در عملکرد عامل است: هنگامی‌که هدف کلیدیِ تعریف‌شده از سوی کاربر (مثلاً حفظ توانایی توان عملیاتی شبکه) با اولویت‌بندی به حداقل رساندن مشکل فقدانِ بسته توأم گردد، مدلِ هوشمند درمی‌یابد که باید به منظور جلوگیری از دست‌افتادگیِ پیام‌ها، به سمتِ مسیرهای قابل اعتمادتری متمایل شود که به صورت ذاتی کندتر هستند. در محیط‌های بهینه‌سازی چندهدفه، چنین مصالحه‌ای نه تنها یک نقطه ضعف و محدودیت به شمار نمی‌رود، بلکه از قضا یک رفتار برآمده و مطلوب از سوی عامل ارزیابی می‌گردد. چرا که در اینجا اولویت سیستم بر تضمینِ رساندن بسته‌ها بیش از حفظ سرعت بوده است و در کاربردهایی که صحت اطلاعات در آن‌ها اهمیت حیاتی دارد (مانند مانیتورینگ زیست‌محیطی یا سیستم‌های کنترل صنعتی) این تصمیم به‌عنوان عاقلانه‌ترین انتخاب مهندسی تلقی خواهد شد.

این رفتار کاملاً با اصول انتخاب بهینه در استراتژی‌های پرتو-اپتیمال (Pareto Optimality) همخوانی دارد. هنگامی‌که مرزهای متضاد انتخاب‌ها بررسی می‌گردند، تنظیمات دارای تأخیرِ بی‌نهایت کم‌تر عموماً ارتباط معناداری با نرخ بالای اتلاف بسته‌ها — به دلیل استراتژی‌های تهاجمی‌ترِ انتقال داده — نشان می‌دهند.

راهکار RS-DRL یاد می‌گیرد این خطِ مصالحه را با ظرافت طی کند و تصمیم‌ها را منطبق بَر بستر نیاز و با تخصیص وزن‌های اختصاص‌یافته برای هر متغیر اخذ نماید.

۶-۱-۳ پیامدهای مهندسی سیستم‌های خودوفقی

ثبات، پایداری و همگرایی سریع متد RS-DRL عوامل حیاتی در محیط‌های پویایی هستند که مقیاس و شرایط آن‌ها به سرعت در حال جابجایی است. از طریق در هم تنیدگی مکانیزم شکل‌دهی پاداش، RS-DRL تضمین می‌نماید که عامل تصمیم‌ساز حتی بدون نیاز به بازآموزی و تخصیص پیکربندی‌های جدید به تغییرات پاسخ می‌دهد، در حالی که در اکثر DRLهای استاندارد الزام به طی مجدد مسیر آموزش برای اداره کردن امور تغییر یافته، مشهود و گریزناپذیر است. این خصوصیت، این مدل را برای سامانه‌های برخط مانند سیستم‌های ابری و شبکه‌های اشیاء که استقرار و بروزرسانی خودکار استراتژی در آن‌ها واجب است، بسیار متمایز می‌سازد.

به بیان دقیق‌تر، پیامدهای عملی این رهیافت برای مهندسین توسعه‌دهنده محیط‌های خودوفقی شامل موارد زیر است:

۱. **کاهش قابل‌ملاحظه سربار عملیاتی:** مدل‌های سنتی به هنگام برخورد با رانش و تغییر شرایط محیطی نیازمند صرف چرخه‌های متعدد بازآموزیِ پیاپی می‌باشند. در حالی که ویژگی‌های تعمیم‌پذیری بالای RS-DRL امکان دور ماندنِ مستمرِ مدل از بروزرسانی‌های مجدد را فراهم می‌سازد که خود علاوه بر صرفه‌جویی شدید محاسباتی باعث جلوگیری از عملکرد ضعیف مدل در طی ادوارِ گذارِ بازآموزی می‌گردد.

۲. **بی‌نیازی از داده‌های برچسب‌دار:** بر خلاف مدل‌هایی همانند ML4EAS یا DLASER+ که اتکای گسترده‌ای به تجمیعِ پیشینِ داده‌ها داشته‌اند، الگوریتم RS-DRL کاملاً آنلاین و درلحظه توسعه می‌یابد. این قابلیت نقطه قوتِ برجسته‌ای در محیط‌هایی محسوب می‌گردد که تهیه اطلاعات پیش‌آموزشی غیرممکن و پیش‌بینی‌ناپذیر است.

۳. **پایداری و کاهش تخریب تدریجیِ کیفیت متناسب با گسترش:** استواریِ معنادارِ کیفیت و عملکرد در مواجهه با نسخه‌ی ۲۰۱۶ ــ گزینه‌ای (نسخه v1)، گسترش ۱۲۹۶ ــ گزینه‌ای (نسخه v1.1) و مقیاس سرسام‌آور ۴۰۹۶ حالت ممکن (نسخه v2) مؤید آن است که این راهکار متناسب با میزان گستردگی شبکه قابل استقرار است. همچنین افزایش مقطع زمانی همگرایی مدل، به نسبتِ پیچیدگیِ فضای جستجو، شیبی کمتر از حالت خطی (Sublinear) دارد.

۴. **تصمیم‌گیریِ شفاف و قابل مدل‌سازی:** رویه‌هایِ رفتارگونه‌ی عامل (همانند پدیده مصالحه در برخورد

با اتلافِ دیتا و زمانِ توقف) به نحو کاملاً معنادار و قابل تعریفی مدل‌گستر و قابل توضیح می‌باشد که این ویژگی به مهندسان قدرت تحلیل مضاعف و قابلیت اطمینانِ بیشتری عرضه می‌نماید.

۶-۲ محدودیت‌ها و تهدیدات اعتبار (Threats to Validity)

مانند تمامی‌رهیافت‌های پژوهشی، متدولوژی معرفی‌شده با محدودیت‌هایی مواجه است که بررسی صریحِ روایی و صحتِ نتایجِ استخراج‌شده، بستر مناسب را برای پژوهش‌های مکملِ آینده تبیین می‌نماید.

۶-۲-۱ اعتبار داخلی (Internal Validity)

۱. **حساسیت نسبت به ابرپارامترها:** فاکتورِ کنترل‌کننده‌ی شکل‌دهیِ تجربیات تصادفی (ρ)، متولی تنظیم نمودن درصدِ اختصاص تجربه‌هایِ ناموفقِ تغییرِ یافته به نمونه‌های فرضیِ مثبت و امیدوارکننده است. با اینکه از طریق کاوش شبکه‌ای جامع (Grid Search) توانستیم مناسب‌ترین مقادیر ممکن برای بهینه‌سازیِ عملکرد یافت کنیم، لیکن این مقادیر می‌توانند به نسبت سایر سیستم‌ها و دامنه‌های متفاوت، دستخوش نوسان باشند. بدیهی‌ست چنانچه پارامتر بسیار بالا وضع گردد موجب اختلال در روند یادگیری گردیده و مقادیر ضعیفِ آن نیز سبب نقصان در عامل فشارِ اکتشاف خواهد گشت.

۲. **نوسانات بذر تصادفیِ تولید ارقام (Random Seed Sensitivity):** با وجود اجرای بیش از ۳۰ دور فرآیندِ آموزش مستقل همراه با صحت‌سنجی مستحکم آماری، ماهیتِ درگیر با احتمالات - چه در پویایی خود محیط و چه در بدنه تنظیم پاداش‌ها - سبب می‌شود تا خروجی‌های هر دَوره آموزشی با درجاتِ کوچکی از انحراف واریانس مواجه گردند. با این وصف، فواصل و حاشیه‌های اطمینان به دست آمده (قید شده در جداول فصل ۴) تبیین‌گر کمیت این نوسانات هستند.

۳. **مفروضات بازده دودویی (Binary Reward Assumption):** یکی از شرایط پیش‌نیازِ به کار رفته در الگوریتم، اتکا بر مفروضات مرتبط با نتایج قطعی پیامدهایِ انطباقی به‌عنوان اهدافِ توفیق/سقوطِ مطلقِ دودویی (یا در مقیاس‌های استاندارد مشابه) است که نمی‌توان آن را به سهولت با کلیه نیازهای سیستم‌های ترکیبیِ غول‌پیکر وفق داد.

۶-۲-۲ اعتبار خارجی (External Validity)

۱. **تخصص حوزه (Domain Specificity):** فرآیند ارزیابی منحصراً بر روی بستر شبکه‌های حسگر بی‌سیم (DeltaIoT) متمرکز بوده است، که تعمیم‌پذیری نتایج به سایر حوزه‌ها را محدود می‌کند. هرچند

پلتفرم مذکور آینه‌ای از مشکلات اساسی سیستم‌های مدرن را به تصویر کشیده و تاییدِ کیفیت کرده است، با این وصف، اعمال الگوریتمِ توسعه‌یافته بر دیگر شبکه‌هایِ توزیع‌یافته، نیازمند اعتبارسنجیِ مستقلی خواهد بود.

۲. **تمرکز بهینه بر شبکه‌های گسسته:** تدوین و صورت‌بندی فعلی به نحوی طراحی گشته که متناسب با فضای تصمیم‌داریِ مدل‌های محدود گسسته پایه‌ریزی شود. لیکن اکثریت سیستم‌های پویا (مانند سیستم‌های مقیاس‌بندی و تنظیمِ ظرفیتِ پردازشی ابری) از پارامترهای پیوسته استفاده می‌نمایند و تنظیم معماری RS-DRL متناسب با متدهای کلاسیک پیوسته نظیر شبکه‌های DDPG ساختارهای بسط یافته مطالعاتی جدیدی را می‌طلبد.

۳. **توسعه‌ی در حوزه‌هایِ حساسِ ایمنی:** استفاده از استراتژی خوش‌بینی بر روی اقداماتی که ذاتاً و در دنیای فیزیکی خطرساز بوده‌اند (همانند پلتفرم‌های هدایت اتومبیل‌های هوشمند و یا سیستم‌های دارورسانی)، بدون در نظر گرفتن شرط احتیاط، ممکن است باعث ترغیب ماشین به انجام فعالیت‌های منجر به حادثه گردد و افزودن حلقه‌های تکمیلی تایید صلاحیت بسیار با اهمیت است.

۶-۲-۳ اعتبار سازه (Construct Validity)

۱. **شفافیت متریک‌ها:** سنجش عملکرد عامل هوش از قبیل بررسی مجانبیِ ثبات کیفیت زمانی تا رسیدن به راندمان مطلوب آستانه، رویه‌هایی پذیرفته شده می‌باشند؛ اما معرفی فاکتورهای سنجشی جایگزین مانند نرخ رنجوری (رگرت) و شاخص پایداری ممکن است ابعاد ناشناخته دیگری را نمایان سازند.

۲. **اصطکاک سنجشی مدل‌های جایگزین:** از آنجا که مدل‌های برگرفته از HER نیازمند تنظیم صریحِ برچسب‌گذاریِ مقصد بوده‌اند، مقایسه صورت پذیرفته نیاز به در نظر گرفتن جایگزین‌های ضمنی کیفی داشت. هرچند مستندسازی اعمال شده بسیار متوازن بود ولی باید اذعان داشت که استفاده از این مقادیرِ تعدیل یافته معادل بهینه‌ای برای تمام ظرفیت موجود این دست از رقبا نمی‌تواند قلمداد گردد.

۶-۳ مسیرهایی برای تحقیقات آینده

۶-۳-۱ استراتژی‌های انطباقی و وفقی شکل‌دهی ρ

همان‌طور که در فصل بررسی چالش‌ها بحث شد، اتکا به متغیر کنترل‌گر ρ برای الگوریتم ضروری است. ما در توسعه‌های آتی در نظر داریم با برنامه‌ریزی یک ساختار هوشمند برای تنظیم نرخ فاکتور کنترل‌کننده متناسب با فرآیندهای زمانی مدل پیشروی کنیم تا قابلیت استقرار در زمان اجرای سیستم را تسهیل نماییم. برخی از جهت‌گیری‌های تخصصی شامل موارد زیر می‌باشد:

۱. **برنامه‌ریزی تنظیمیِ پویای پارامتر (ρ):** به جای یک متغیر تنظیم شده‌ی دستی، از سازوکاری جهت ارتقا یا تخفیفِ متغیر بهره خواهیم جست؛ بدین گونه که در هنگامه موفقیت‌های مکرر و تسلط ماشین (فاز استثمار) از درصدِ فاکتور ρ کاسته شده و در هنگام درگیری مستمرِ ماشین و گرفتار شدن آن درون تله‌ی بهینه محلی که نرخ دریافت ارزش با فُتول روبرو می‌گردد به صورت پویا با ارتقای متغیر ضریب فشارِ فرار و اکتشاف تجربیات جدید تشدید شود. در این مسیر استفاده از مدل‌های فرا_یادگیری (Meta-Learning) نیز در خصوص استنتاج بهینه فاکتور مفید فایده خواهد بود.

۲. **شکل‌دهی هوشمند بر پایه‌ی خطای عدم‌اطمینان:** امکان به‌کارگیری ابزارهای ارزیابی تخمین اطمینان (مانند Dropout یا انواع مدل‌های اِنسامبل) در الگوریتم، مقدور می‌سازد تا آن دسته از تجربه‌های شکست خورده‌ای در چرخه‌ی تصادفی‌سازی اولویت یابند که سیستم درباره‌ی قطعیت خطاهای‌شان کمترین اطمینان و شناخت را در محیطِ غیرشناخته از خود نشان داده بود.

۳. **اولویت‌بندی حافظه‌ی نمونه‌گیری:** ادغام روش انتخاب هوشمند با الگوریتم‌های حافظهِ مبتنی بر اولویت (Prioritized Experience Replay)، عاملی را معرفی می‌کند تا زیرمجموعه‌های مردود شده‌ی مستعدِ جبران پس از اعمال شکل‌دهی، با فرکانس شانسِ بیشتری برای بازیابی توسط شبکه مجدداً احضار شوند.

۶-۳-۲ گسترش به دامنه‌های پیوسته و حساس به ایمنی

با عنایت به ماهیت پویای فضای نرم‌افزار، توسعه روش‌های حاضر بر مسائل نوین‌تر ضروری خواهد بود:

۱. **فضای اعمال پیوسته (Continuous Action Spaces):** بسط معماری متدیک فعلی در بَطنِ متدهایِ قدرتمندتری نظیر سیستم‌های شیبِ سیاستی همانند متدهای PPO و SAC از اهداف توسعه‌ی اصلی

پژوهش است تا علاوه بر شکل‌دهی پاداش تصادفی، بتوان پارامترهای بدون گسستگی (همچون تنظیم منابع سخت‌افزاری پردازش در فضای ابری) را تنظیم نمود.

۲. **اعمال محدودیت‌های تایید ایمنی (Safe Reshaping):** برای رفع نگرانی مطرح شده در دامنه‌های حساس به ایمنی، در نظر داریم مکانیزمی‌با عنوان «شکل‌دهی ایمن» پدید آوریم که بازتولیدِ تصادفیِ نتایج شکست‌خورده تنها زمانی مُیسّر گردد که خط‌مشی پیش‌بینی‌شده در مناطق ایمن تاییدشده قرار گیرد. ترکیب با ابزارهای اعتبارسنجی صوری (همچون UPPAAL-SMC استفاده شده در این تحقیق) از هدایت به وضعیت‌های ناایمن پیشگیری می‌کند.

۳. **گسترش به سامانه‌های چند عاملی (Multi-Agent):** برای مقابله با فضاهایی با پویاییِ بخش‌های تطبیقی در حال تعامل با هم، گسترش متد در چارچوبی صورت می‌پذیرد که چند عامل در هماهنگی استراتژی‌های کاوش همسان به شکل‌دهی پاداش‌های منسجم برای تحقق اهداف بزرگ عمل کنند.

۶-۳-۳ اعتبارسنجی بین‌دامنه و تحلیل‌های نظری

جهت بررسی فراگیری این استراتژی و ایجاد پایه‌های نظری قوی‌تر اهداف زیر دنبال خواهد گردید:

۱. **پیکربندی شبکه‌های ابری (Cloud Auto-scaling):** به‌کارگیری این رویکرد به جهت تخصیص و توزیع مقیاس ماشین‌های مجازی، و ارزیابی در برابر کارهای مرتبط دیگر در این محیط.

۲. **هماهنگی وسایل نقلیه خودران (Autonomous Vehicles):** آزمایش در محیط‌های شبیه‌سازی برای هماهنگ‌سازی و تطبیق الگوهای ترافیکی با محدودیت‌های ایمنی و اهداف چندگانه (زمان سفر، بهینه‌سازی انرژی و امنیت).

۳. **تضمین تئوریک فرآیند همگرایی (Theoretical Analysis):** بررسی اینکه آیا سیگنال پاداش شکل‌دهی‌شده ویژگی انقباضی و نگاشت ثابت برای یادگیری کیو (Q-Learning) را حفظ می‌کند، و محاسبه کران‌های خطای احتمالی انحرافِ ارزش Q نسبت به سیاست بهینه.

۴. **کاهش حجمِ پیچیدگی نمونه‌ها (Sample Complexity Analytical Analysis):** کمیت‌بندی تئوریک پیرامون اثربخشیِ شکل‌دهی پاداش بر تقلیل شمار نمونه‌های لازم برای دستیابی به خروجی‌های نزدیک به بهینه در قیاس با شبکه‌ی DQN استاندارد.

۶-۴ نتیجه‌گیری

۶-۴-۱ خلاصه بیان مسئله و راهکار پیشنهادی

این رساله به‌عنوان دستاوردی متصل بر یکی از چالش‌های بنیادین سیستم‌های خودوفقی مقاصد خود را تنظیم نموده است؛ مدیریت عدم‌قطعیت در شبکه‌هایی که درون فضاهای بزرگ و گسسته‌ی تطابق عمل می‌کنند. رویکردهای سنتی - اعم از مدل‌های مبتنی بر یادگیری ماشین، مدل‌های متکی بر جستجو و یا یادگیری تقویتی استاندارد - با محدودیت‌های شدیدی مواجه بوده‌اند؛ ویژگی‌هایی از قبیل نیاز مبرم به دیتای برچسب‌دار آفلاین، دست و پنجه نرم کردن با فقدان مقیاس‌پذیری و احتمال بالای توقف در بهینه‌های محلیِ برآمده از تغییر شرایط.

ما استراتژی **یادگیری تقویتی عمیق با شکل‌دهی تصادفی پاداش (RS-DRL)** را ارائه نمودیم؛ رویکردی نوین که مکانیسم تصادفیِ شکل‌دهی را درون جریان حافظه بازپخشِ عامل کیو (DQN) یکپارچه می‌نماید. با بهره‌گیری از زیرمجموعه‌ای از تجربیات ناکامِ فاز گذشته ($r_i = ۰$) و اعمال تحول در پاداش و تبدیل آن به تجربه‌ای خوش‌بینانه ($r_i = ۱$)، راهکار RS-DRL مشوق قدرتمندِ فرآیندِ اکتشاف بوده و از گرفتاری در بهینه‌های محلی جلوگیری می‌نماید، بدون آنکه کمترین الزامی‌برای تعریف و اختصاص قطعیِ اهدافِ مبتنی بر دانش از پیش‌تعیین‌شدهِ دامنه، داشته باشد.

۶-۴-۲ مرور مشارکت‌ها و شواهد تجربی

یافته‌های مستخرجِ پژوهش که ارائه‌ی پاسخی قاطع بر پرسش‌های این رساله به شمار می‌آید شامل موارد کلیدی زیر است. جدول ۶-۱ خلاصه‌ای از این مشارکت‌ها و شواهد را نشان می‌دهد.

جدول ۶-۱: خلاصه‌ای از مشارکت‌های کلیدی و شواهد تجربی

مشارکت (Contribution)	شواهد (Evidence)	اهمیت (Significance)
شکل‌دهی پاداش نوین (Novel Reward Shaping)	توسعه الگوریتم ۴-۱، نمایش در شکل‌های ۵ و ۶	امکان اکتشاف بدون نیاز به دانش دامنه سیستم
تطبیق و وفقی‌سازی مقیاس‌پذیر	مستخرج از جداول ۳ و ۴ (نسخه‌های v1 تا v2)	حفظ پایداریِ عملکرد با رشد گزینه‌ها از ۲۱۶ تا ۴۰۹۶ حالت
برتری و تفوق اثبات آماری	تحلیل جداول ۵ و ۶، معناداری ۲۵ از ۲۸ تست	دریافت $p < ۰/۰۰۱$ برای اکثر متریک‌های بهبود اتلاف بسته

مشارکت اول: ابداع ماشین نوین جهت شکل‌دهی پاداش توسعه الگوریتم تخصیص پاداش (طراحی الگوریتم ۴-۱) که از طریق کنترل‌کننده‌ای با ضریب مشخصِ (ρ) عمل نموده و برخلاف رویکردهای غایت‌محور نظیر متدهای HER هیچگونه نیازی بر تعیین هدف آستانه‌ای با درک معنایی نداشته و مستقیماً روی سیستم‌های خودوفقی قابل راه‌اندازی است.

مشارکت دوم: یکپارچه‌سازی حلقه کنترل MAPE-K طراحی زیرِسامانه‌ی متصل متشکل از چرخه ۹ مرحله‌ای که RS-DRL را با حلقه MAPE-K ترکیب می‌کرد؛ مجهز به ارزیابی برای تضمین اعتبارِ زمانِ اجرا با موتور کنترل UPPAAL-SMC و مؤلفه ActivFORMS برای اجرای وفقی.

مشارکت سوم: دستاوردهای تجربی در مقیاس اینترنت اشیاء دستاورد استقرار شبکه‌ی عامل بر بستر ۳ مقیاس وسیع (فضای محدود ۲۱۶، نسخه ۱۲۹۶ و نسخه منبسط ۴۰۹۶ تصمیم ممکن). دستیابی بی‌رقیب به نرخ مجانبی ۰/۰۲۶۸ ± ۰/۹۹۰۵ (در شبکه اولیه) و ۰/۰۹۳۲ ± ۰/۸۹۸۳ (در v1.1) که با قاطعیت ساختار کلاسیک ϵ-greedy DQN (با راندمان ۰/۲۲۳۳ ± ۰/۴۵۵۹) و مشابه‌های HER را مغلوب ساخت. ثبت کاهش شگفت‌انگیزِ ۲۲/۲۳٪ برای نابودی بسته‌ها و ۰۷/۳۹٪ کاهش تأخیر شبکه نسبت به DLASER+ تحت اهداف TTS. این مقادیر از طریق نتایج آزمون t-test (موفقیت در ۳.۸۹٪ از ۲۸ مقایسه متمایز) اعتبار مطلق دریافت کرد.

مشارکت چهارم: بینش عمیق نظری و عملی اثبات و آشکارسازیِ اینکه مکانیزمِ روایی برای تزریق خوش‌بینی نه تنها راه‌حلی منطقی، خروج سیستم را از گیر افتادن مصون داشته، بلکه تحلیل الگوهای متضاد درون‌سیستمی‌نظیر (تأخیر متناقض با نرخ اتلاف بسته‌ها) نشان دادیم عامل استنتاج‌هایی به وسعت دانش طراحی چند_هدفه و سازگار با رفتار سیستم هوشمند اخذ کرده است و همزمان این ظرفیت برای تغییر سایز و پدیده‌ی مقیاس‌پذیری فضای عمل بسیار کارآمد بوده است.

۶-۴-۳ سخنان پایانی

از آنجا که سیستم‌های نرم‌افزاری به‌طور فزاینده‌ای در محیط‌هایی سرشار از عدم قطعیت، پویایی و مقیاس‌های بزرگ عمل می‌کنند، نیاز به مکانیسم‌های وفقی که بتوانند به‌طور کاملاً مستقل فرآیند یادگیری و تنظیم استراتژی را انجام دهند چشمگیرتر می‌شود. مدل هوشمند RS-DRL یک قدم استوار در جهت این چشم‌انداز تلقی می‌شود - روشی که نه‌تنها متکی به آموزش‌های مبتنی بر موفقیت است بلکه با ظرافتی شایسته، از بطن اشتباهات پیشین درس‌های بسیار موثری اخذ می‌کند و به بیان دیگر کاستی‌های پیشین را به یک فرصت ارزنده تبدیل خواهد کرد.

اهمیت این تحقیقات مرزهای توسعه‌یافته در یک شبکه DeltaIoT را کاملاً در هم شکسته است. هر نوع معماری که توأمان با ۳ چالش بنیادین سیستمِ نرم‌افزار (استقرار محیط پیکربندی کلان، عدم‌قطعیت‌های متغیر شرایط و نیاز تطبیق زمانِ اجرا) دست و پنجه نرم کند، با تکیه بر متد معرفی‌شده در RS-DRL می‌تواند بر این معضل غلبه نماید، به گونه‌ای که سیستم‌های ابری، امنیتِ سایبریِ شبکه، کنترلِ برق هوشمند و هدایت پهپادهای یکپارچه از مصادیق این قابلیت‌ها هستند.

مأموریتِ نهایی آینده برای سیستم‌ها آن است که نه تنها مستمراً راهبریِ سیستم را هماهنگ بدانند بلکه فراتر از آن استراتژی را پیوسته غنی‌سازی نمایند — هدفی بزرگ که در شُرف تحقق پیوسته است. رساله‌ی حاضر با اتکا بر راهکار ابداعی RS-DRL تلاش داشته خشتِ کوچکی اما ارزشمند رو به سوی این آرمان کلان مهندسی نرم‌افزار بنا کند.

کتاب‌نامه

[1] J. O. Kephart and D. M. Chess, "The vision of autonomic computing," *Computer*, vol.36, no.1, pp.41–50, 2003.

[2] P. Oreizy, N. Medvidovic, R. N. Taylor, D. S. Rosenblum, J. R. Robbins, M. Mikic-Rakic, N. Lekic, R. Webby, and N. Medvidovic, "An architecture-based approach to self-adaptive software," *IEEE Intelligent Systems*, vol.14, no.3, pp.54–62, 1999.

[3] B. H. Cheng, R. de Lemos, H. Giese, P. Inverardi, J. Magee, *et al.*, "Software engineering for self-adaptive systems: A research roadmap," in *Software Engineering for Self-Adaptive Systems*, pp.1–26, Springer, 2009.

[4] F. Quin *et al.*, "Efficient analysis of large adaptation spaces in self-adaptive systems using machine learning," in *2019 IEEE/ACM 14th International Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS)*, IEEE, 2019.

[5] F. Quin, D. Weyns, and O. Gheibi, "Reducing large adaptation spaces in self-adaptive systems using machine learning," *Journal of Systems and Software*, p.111341, 2022.

[6] O. Gheibi and D. Weyns, "Dealing with drift of adaptation spaces in learning-based self-adaptive systems using lifelong self-adaptation," *ACM Transactions on Autonomous and Adaptive Systems*, vol.19, no.1, pp.1–57, 2024.

[7] Z. Coker, D. Garlan, and C. L. Goues, "Sass: Self-adaptation using stochastic search," in *10th International Symposium on Software Engineering for Adaptive and Self-Managing Systems, SEAMS 2015*, Institute of Electrical and Electronics Engineers Inc, 2015.

[8] C. Kinneer *et al.*, "Managing uncertainty in self-adaptive systems with plan reuse and stochastic search," in *ACM/IEEE 13th International Symposium on Software Engineering for Adaptive and Self-Managing Systems, SEAMS 2018, , co-located with International Conference on Software Engineering, ICSE 2018*, IEEE Computer Society, 2018.

[9] C. Kinneer, D. Garlan, and C. L. Goues, "Information reuse and stochastic search: Managing uncertainty in self-* systems," *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, vol.15, no.1, pp.1–36, 2021.

[10] D. Kim and S. Park, "Reinforcement learning-based dynamic adaptation planning method for architecture-based self-managed software," in *2009 ICSE Workshop on Software Engineering for Adaptive and Self-Managing Systems*, IEEE, 2009.

[11] A. Metzger *et al.*, "Realizing self-adaptive systems via online reinforcement learning and feature-model-guided exploration," *Computing*, pp.1–22, 2022.

[12] M. L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, 1994.

[13] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 2018.

[14] G. Du, Y. Wang, Y. Liu, X. Wu, and L. Ma, "Alleviating local optima and enhancing path planning: A deep reinforcement learning approach for autonomous exploration," in *International Conference on Autonomous Unmanned Systems*, (Singapore), pp.124–133, Springer Nature Singapore, September 2023.

[15] I. Gerostathopoulos, D. Weyns, S. Bogo, *et al.*, "Deltaiot: A benchmark for self-adaptive iot systems," in *Proceedings of the 14th International Symposium on Software Engineering for Adaptive and Self-Managing Systems*, pp.66–72, 2019.

[16] F. Kachi and C. Bouanaka, "A hybrid model for efficient decision-making in self-adaptive systems," *Information and Software Technology*, vol.153, p.107063, 2023.

[17] C. Kinneer *et al.*, "Building reusable repertoires for stochastic self-* planners," in *2020 IEEE International Conference on Autonomic Computing and Self-Organizing Systems (ACSOS)*, IEEE, 2020.

[18] T. Chen *et al.*, "Femosaa: Feature-guided and knee-driven multi-objective optimization for self-adaptive software," *ACM Transactions on Software Engineering and Methodology*, vol.27, no.2, 2018.

[19] J. Cámara, H. Muccini, and K. Vaidhyanathan, "Quantitative verification-aided machine learning: A tandem approach for architecting self-adaptive iot systems," in *2020 IEEE International Conference on Software Architecture (ICSA)*, IEEE, 2020.

[20] Y. Du *et al.*, "Scalability and robustness of deep reinforcement learning in large discrete action spaces," *IEEE Transactions on Neural Networks and Learning Systems*, 2023.

[21] M. Andrychowicz, F. Wolski, A. Ray, J. Schneider, R. Fong, P. Welinder, B. McGrew, J. Tobin, P. Abbeel, and W. Zaremba, "Hindsight experience replay," in *Advances in Neural Information Processing Systems*, vol.30, 2017.

[22] Q. He, L. Zhuang, and H. Li, "Soft hindsight experience replay," *arXiv preprint arXiv:2002.02089*, 2020.

[23] A. Y. Ng, D. Harada, and S. Russell, "Policy invariance under reward shaping: Theory and application to relation preservation," in *ICML*, vol.99, pp.278–287, 1999.

[24] R. Calinescu, C. Ghezzi, M. Kwiatkowska, and R. Mirandola, "Formal verification of self-adaptive software systems at runtime," *Journal of Systems and Software*, vol.84, no.12, pp.2223–2239, 2011.

[25] D. Weyns *et al.*, "Formal models for self-adaptation: a systematic literature review," *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, vol.7, no.1, pp.1–36, 2012.

[26] D. Weyns, M. U. Iftikhar, *et al.*, "Activforms: Active formal models for self-adaptation," *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, vol.13, no.3, pp.1–41, 2018.

[27] A. David *et al.*, "Uppaal smc tutorial," *International journal on software tools for technology transfer*, vol.17, pp.397–415, 2015.

پیوست آ

جریان‌های کاری کامل فرآیند

پیوست ب

لیست کامل الگوریتم‌ها

پیوست پ

نتایج تجربی گسترده

پیوست ت

مصنوعات پیاده‌سازی و پکیج بازتولید

Abstract:

Contemporary software systems increasingly operate in dynamic environments characterized by significant uncertainties, such as network interference and traffic fluctuations, which can jeopardize Quality of Service (QoS) objectives. Self-adaptive systems (SAS) provide a fundamental approach to address these challenges using feedback loops. However, managing large and discrete adaptation spaces (comprising hundreds or thousands of configurations) remains a significant challenge. Existing approaches, including machine learning and classical reinforcement learning, often suffer from poor scalability, reliance on labeled data, or being trapped in local optima under environmental shifts. This thesis proposes and evaluates a novel method called Randomized Reward-Shaped Deep Reinforcement Learning (RS-DRL) for efficient uncertainty management in SAS. The proposed method integrates a Deep Q-Network (DQN) core within the standard MAPE-K control loop. The primary innovation is the "Randomized Reward Shaping" mechanism, which strategically redesigns a random subset of failed past experiences in the replay buffer with optimistic (pseudo-success) rewards. Crucially, the reward signal is gated by formal verification results from UPPAAL-SMC, ensuring that learning is guided by formal safety and quality constraints. While the formal model requires domain-specific configuration, the randomized shaping heuristic itself is domain-agnostic and does not require explicit goal coordinates or intensive domain expertise for exploration. Experimental evaluation was conducted on the DeltaIoT exemplar within wireless sensor network scenarios featuring adaptation spaces of up to 4096 configurations. The results demonstrate that RS-DRL significantly outperforms state-of-the-art baselines, including DQN, HER, Soft HER, and DLASER+. Specifically, under the Threshold-based (TT) goal, RS-DRL achieved a 23.22% reduction in packet loss and a 39.07% reduction in latency compared to the DLASER+ baseline. Statistical analysis confirms the robustness and scalability of the proposed approach, establishing RS-DRL as a resilient optimization strategy for next-generation complex self-adaptive systems.

Keywords: Self-Adaptive Systems, Deep Reinforcement Learning, Randomized Reward Shaping, Uncertainty Management, Formal Verification, UPPAAL-SMC

Tarbiat Modares University
Faculty of Electrical and Computer Engineering

Randomized Reward-Shaped Deep Reinforcement Learning for Managing Uncertainty in Self-Adaptive Systems

A Thesis Submitted in Partial Fulfillment of the Requirement for the Degree of Doctor of Philosophy in Computer Engineering

By:

Alireza Kavianifar

Supervisor:

Dr. ...

Advisor:

Dr. ...

February 2026