

مدل سازی عملیات مونتاژ خودکار رباتیک توسط ماتریس های حالت

سید مجید نورحسینی

دانشکده مهندسی کامپیوتر و فن آوری اطلاعات،

دانشگاه صنعتی امیرکبیر

majidnh@aut.ac.ir

عطیه میرشاه‌ولد

دانشکده مهندسی کامپیوتر و فن آوری اطلاعات،

دانشگاه صنعتی امیرکبیر

mirshahvalad_a@aut.ac.ir

استفاده شده است که شامل مدل سازی جهان مسئله و مدل سازی عملگرهای سیستم طرح‌ریزی می باشد. این روش که بازنمایی ماتریس حالت نام دارد همراه با یک الگوریتم عمومی طرح‌ریزی در [۸] معرفی شده است. در اینجا، برخی مفاهیم ارائه شده در مقاله مذکور را مرور کرده و سپس کاربرد این تکنیک را در طرح‌ریزی عملیات مونتاژ خودکار رباتیک نشان خواهیم داد.

۲- بازنمایی مسئله مونتاژ

ملاحظه اصلی در مونتاژ این است که سیستم در حین مونتاژ کردن ساختار هدف در طی فرآیند مونتاژ به تدریج از محدودیت‌های پیکربندی اولیه می‌کاهد. بدین صورت که سیستم ابتدا سعی می‌کند تا اگر ضرورت دارد ساختار بخش اولیه را از هم جدا کند و سپس سعی در ساخت بخش هدف دارد. فرض کنید اجزایی که در یک عمل مونتاژ دخیل هستند به صورت زیر باشند:

$$E = \{e_1, e_2, \dots, e_N\};$$

و فرض کنید مجموعه ارتباطات بروی این اجزاء به صورت زیر است:

$$L = \{l_r(E') \mid r = 1, 2, \dots, M; E' \subseteq E; M \leq N\}$$

در معادله فوق $l_r(E')$ بدین معنی است که عناصر موجود در E' رابطه l_r را ارضا می‌کنند.

مجموعه ارتباطات L می‌تواند از طریق یک ماتریس مجاورت که در این جا آن را ماتریس حالت می‌نامیم به صورت زیر نشان داده شود:

$$S = [s_{i,j}], \quad i, j = 1, 2, \dots, N$$

where

$$s_{i,j} = \begin{cases} r & \text{if } l_r(e_i, e_j) \in L \\ 0 & \text{otherwise} \end{cases}$$

مقدار صفر به این معنی است که هیچ ارتباط مستقیمی بین دو بخش وجود ندارد. در کل، رابطه r ، یک رابطه در L را به عددی از مجموعه

$$r: L \rightarrow Q \text{ یعنی } Q = \{1, 2, \dots\}$$

یک مسئله مونتاژ شامل دو مجموعه از ارتباطات ها است:

- مجموعه ارتباطات ساختار بخش اولیه:

چکیده: در این مقاله ابتدا روشی جدید بنام بازنمایی ماتریس حالت^۱ برای مدل سازی مسئله مونتاژ معرفی گردیده و سپس بر مبنای این بازنمایی، الگوریتمی جهت طرح‌ریزی^۲ خودکار ربات ارایه می‌شود. بازنمایی ماتریس حالت در مسئله مونتاژ، از طریق محدودیت‌های وضعیت اولیه و ساختار وضعیت هدف شکل می‌گیرد. از محاسن این رویکرد آن است که به کمک این روش، الگوریتم طرح‌ریزی رباتیک آسان شده و می‌تواند بسادگی و با کارایی مناسب، از طریق عملیات ساده ماتریسی پیاده‌سازی شود. در ادامه، چگونگی طرح‌ریزی خودکار یک نمونه مسئله مونتاژ را بر مبنای این روش خواهیم دید.

واژه های کلیدی: طرح‌ریزی خودکار عملیاتی ربات، مسئله مونتاژ، بازنمایی ماتریس حالت.

۱- مقدمه

یکی از جنبه های یک ماشین هوشمند که قادر به اجرای خودمختار یک کار^۳ باشد، توانایی آن در طرح‌ریزی خودکار می باشد. در حوزه رباتیک، طرح‌ریزی اغلب به یک سلسله مراتب هرمی متشکل از سطوح مختلف یک سیستم رباتیک، تجزیه می‌شود که می‌تواند از سطوح پایین طرح‌ریزی خط سیر^۴ تا بالاترین سطح که در آنها طرح‌ریزی به معنی ساماندهی مجموعه ای از اعمال برای رسیدن به یک هدف است مد نظر باشد. هدف این تحقیق، طرح‌ریزی در بالاترین سطح این سلسله مراتب هرمی است. طرح‌ریزی در این سطح به علت مسائل ترکیبی^۵ زیادی که باید قبل از آن که در یک طرح جهان واقعی بتواند فرموله شود حل شوند، بسیار پیچیده است. [۹]

بصورت کلاسیک، انواع مختلف مسندات^۶ برای بازنمایی یک مسئله طرح‌ریزی استفاده می‌شدند و الگوریتم های طرح‌ریزی که در طرح‌ریزنده های کلاسیک ارایه می‌شد به دستکاری این مسندات نیاز داشتند. در این مقاله روشی جدید برای بازنمایی مسئله طرح‌ریزی

1. state matrix representation

2. planning

3. task

4. trajectory planning

5. combinatorial

6. predicate

تعریف می شوند که به ترتیب ماتریس عمل حذف کننده^{۱۱} و ماتریس عمل برقرار کننده^{۱۲} هستند. H_n به صورت زیر تعریف می شود:

$$H_n = [h_{i,j}], \quad i, j = 1, 2, \dots, N \quad \text{where}$$

$$h_{i,j} = \begin{cases} \alpha_r n(e_i, e_j) & \text{if } S[i, j] = r \text{ and } i \neq j \\ 0 & \text{otherwise} \end{cases}$$

H_n شامل تمامی اعمالی است که برای انتقال وضعیت S به یک وضعیت $null$ مورد نیاز است. به طور مشابه ماتریس عمل برقرار کننده H_p بروی یک وضعیت S به صورت زیر بیان می شود:

$$H_p = [h_{i,j}], \quad i, j = 1, 2, \dots, N \quad \text{where}$$

$$h_{i,j} = \begin{cases} \alpha_r p(e_i, e_j) & \text{if } S[i, j] = r \text{ and } i \neq j \\ 0 & \text{otherwise} \end{cases}$$

هر عمل h با یک ماتریس غالب^{۱۳} F مربوط است که شامل شرایط غالب^{۱۴} یک عمل می باشد. ماتریس غالب یک عمل، وضعیتی را مشخص می کند که در آن، عمل می تواند انجام شود. تفاوت بین شرایط غالب یک عمل و پیش شرایط و پس شرایط متداول^{۱۵} اختصاص یافته به یک عمل این است که شرایط غالب آن شرایطی هستند که برای اجرای یک عمل ضروری می باشد اما از اجرای عمل تأثیر نمی پذیرند. به عبارت دیگر، اینها پیش شرایطی هستند که در پس شرایط اعمال، ثابت باقی می ماند. به عنوان یک بیان رسمی، برای یک عمل $h(e_i, e_j)$ ماتریس غالب $F(h_{i,j})$ یک ماتریس $N \times N$ است که بصورت زیر تعریف می شود:

$$F(h_{i,j}) = [f_{k,l}], \quad i, j = 1, 2, \dots, N \quad \text{where}$$

$$f_{k,l} = u \text{ unless determined otherwise } k, l = 1, 2, \dots, N$$

در رابطه بالا منظور از u مقدار تعریف نشده یا همان *don't care* است. در کل، ماتریس غالب یک عمل، بوسیله توابع و مشخصات سیستم مونتاژ تعیین می شود. تکنیک های سیستم خبره نیز ممکن است برای پیدا کردن شرایط غالب در یک عمل مونتاژ، استفاده شوند. در برخی دامنه ها، ماتریس های غالب می توانند بسادگی فرمول بندی شوند.

تعریف ۱:

یک ماتریس حالت نظیر S_1 ، ماتریس حالت دیگری مثل S_2 را ارضا می کند و به صورت $S_1 \sqsubseteq S_2$ نمایش داده می شود اگر و تنها اگر

$$\forall i, j \quad S_1[i, j] = u \quad \text{or} \quad S_1[i, j] = S_2[i, j]$$

$$L_0 = \{l_{r_1}(E') \mid r_1 = 1, 2, \dots, j; E' \subseteq E; j \leq N\}$$

• مجموعه ارتباطات ساختار بخش هدف:

$$L_* = \{l_{r_2}(E') \mid r_2 = 1, 2, \dots, K; E' \subseteq E; k \leq N\}$$

به این ترتیب ماتریس حالت اولیه S_0 بصورت زیر است:

$$S_0 = [a_{i,j}], \quad i, j = 1, 2, \dots, N \quad \text{where}$$

$$a_{i,j} = \begin{cases} r & \text{if } l_r(e_i, e_j) \in L_0 \\ 0 & \text{otherwise} \end{cases}$$

و ماتریس حالت هدف S_* بصورت زیر می باشد:

$$S_* = [b_{i,j}], \quad i, j = 1, 2, \dots, N \quad \text{where}$$

$$b_{i,j} = \begin{cases} r & \text{if } l_r(e_i, e_j) \in L_* \\ 0 & \text{otherwise} \end{cases}$$

برای تحقق هر ارتباطی و نیز برای قطع آن، یک عمل سیستمی وجود دارد. از اینرو دو عمل برای هر ارتباط لازم است: یک عمل برقرار کننده^۷ و دیگری عمل حذف کننده^۸. عمل برقرار کننده، ارتباط را زمانیکه مطلوب است برقرار می کند. عمل حذف کننده، ارتباط را زمانیکه به عنوان یک محدودیت برآورده شد، قطع می کند. در یک قالب رسمی دو عمل برای ارتباطی مثل $l_r(e_i, e_j)$ بدین صورت می باشد: عمل حذف کننده با $\alpha_r n(e_i, e_j)$ و عمل برقرار کننده با $\alpha_r p(e_i, e_j)$ نمایش داده می شود. عمل حذف کننده رابطه r بین e_i و e_j را قطع می کند و یک ارتباط $null$ بروی e_i ایجاد می کند و عمل برقرار کننده رابطه r را بین e_i و e_j برقرار می کند. همچنین یک عمل ترکیبی^۹ بدین صورت تعریف می شود:

$$\alpha_{r_1, r_2}(e_i, e_j, e_k)$$

که به معنی، قطع ارتباط r_1 بین قطعه e_i و e_j و برقراری ارتباط r_2 بین قطعه e_i و e_k می باشد. یک عمل ترکیبی نظیر $\alpha_r(e_i, e_j, e_j)$ در واقع نمایش گر "بی عملی" است. چون ارتباط r بین e_i و e_j را قطع می کند و بلافاصله دوباره همین ارتباط را بین همان قطعات برقرار می کند. متناظر با آنچه در مورد بازنمایی پیکر بندی بوسیله ماتریس S داشتیم، دو ماتریس H_p و H_n نیز

¹¹.eliminating action matrix

¹².establishing action matrix

¹³.prevail matrix

¹⁴.prevailing precondition

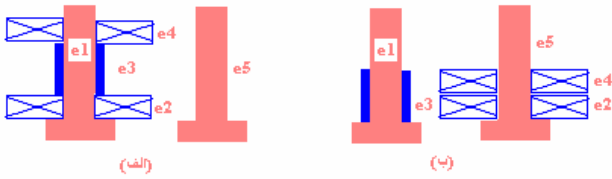
¹⁵.conventional pre- and post condition

⁷.establishing action

⁸.eliminating action

⁹.combo action

¹⁰.no-action



شکل (۱). یک مسئله مونتاژ: (الف) وضعیت اولیه و (ب) ساختار هدف است.
گام ۱: قدم اول، بازنمایی مسئله مونتاژ به کمک ماتریس های حالت است. مجموعه اجزای درگیر در مسئله چنین است:

$$E = \{e_1, e_2, e_3, e_4, e_5\};$$

در این مثال از بازنمایی ماتریس حالت ساده شده استفاده می کنیم. البته این ساده سازی لطمه ای به عمومیت روش نمی زند. ماتریس های اولیه و هدف بصورت زیر می باشند:

$$S_0 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad S_* = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

اعمال مورد نیاز جهت انتقال از وضعیت اولیه به وضعیت هدف اینها هستند:

$$H = \{an_{2,1}, an_{3,1}, an_{4,1}, ap_{2,5}, ap_{3,1}, ap_{4,5}\}$$

ماتریس های غالب بر این اعمال، با ملاحظات زیر مشخص می شوند:

- برای برداشتن یا نصب e_3 ، قطعه e_4 نمی تواند روی e_1 قرار داشته باشد، بنابراین:

$$F(an_{3,1})[4,1] = F(ap_{3,1})[4,1] = 0$$

- برای برداشتن e_2 ، قطعات e_3 و e_4 نمی توانند روی e_1 قرار داشته باشند. همچنین برای نصب e_2 بروی e_5 ، e_4 ،

نمی تواند روی e_5 قرار داشته باشد، چون راه e_2 را مسدود می کند. در نتیجه:

$$F(an_{2,1})[3,1] = F(an_{2,1})[4,1] = F(ap_{2,5})[4,5] = 0$$

- محدودیتی برای برداشتن یا نصب e_4 وجود ندارد. تمامی دیگر عناصر ماتریس های غالب برابر u (don't care) خواهند بود.

گام ۲: تولید ترتیب جزئی بروی اعمال. عمل حذف کننده بروی یک قطعه، باید قبل از عمل برقرار کننده ای که بروی همان قطعه عمل می کند قرار گیرند. از اینرو داریم:

$$R = \{an_{2,1} \rightarrow ap_{2,5}, an_{3,1} \rightarrow ap_{3,1}, an_{4,1} \rightarrow ap_{4,5}\}$$

فرض کنید H ، یک ماتریس عمل روی حالت S باشد. عملی نظیر $h \in H$ در حالت S قابل اجرا است اگر و تنها اگر $F(h) \subseteq S$ باشد. به بیان دیگر، ماتریس عمل H را، تنها وقتی می توان به ماتریس حالت S اعمال کرد که شرایط غالب مطلوب فراهم باشد. همچنین می توان یک فرم ساده شده از مسئله مونتاژ را به ترتیب زیر تعریف کرد: یک ماتریس حالت ساده شده S_s بروی یک مجموعه ارتباط L ، به صورت زیر تعریف می شود:

$$S_s = [s_{i,j}], \quad i, j = 1, 2, \dots, N \quad \text{where}$$

$$s_{i,j} = \begin{cases} 1 & \text{if } l_r(e_i, e_j) \in L \\ 0 & \text{otherwise} \end{cases}$$

در رابطه فوق مقدار صفر به معنی عدم ارتباط و مقدار یک به معنی وجود ارتباط (هر ارتباطی) بین دو قطعه می باشد. در فرم ساده شده، $s_{i,j}$ یک مقدار دودویی اختیار می کند و اعمال حذف کننده، برقرار کننده و ترکیبی به ترتیب با $an_{i,j}$ ، $ap_{i,j}$ و $al_{i,j,k}$ نشان داده می شود.

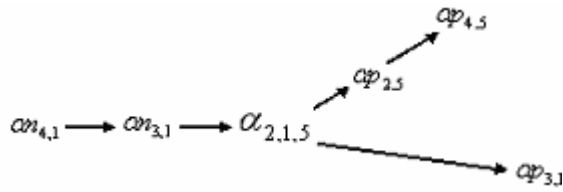
۳- الگوریتم طرح ریزی

الگوریتم طرح ریزی شامل سه گام است. گام اول مربوط به بازنمایی مسئله مونتاژ است. گام دوم یک طرح را تولید می کند که متشکل از ترتیب های جزئی بروی اعمال حذف کننده و برقرار کننده است و گام سوم اعمال موجود در شبکه اعمال تولید شده را در هر کجا که ممکن باشد با هم ترکیب می کند و طرح نهایی را تولید می کند.

ورودی طرح ریزنده شامل ماتریس های حالت اولیه و هدف می باشد. طرح ریزنده تمامی اعمال مورد نیاز را تولید و آنها را برای تولید طرح مرتب می کند. فرض می شود که شرایط غالب اعمال در یک پایگاه داده در دسترس هستند. همان طور که پیشتر اشاره شد، این شرایط از طریق بررسی مشخصات سیستم مونتاژ تعیین می شوند. ترتیب ها در یک مجموعه R ذخیره می شوند که در نهایت شامل طرح خواهد بود. از علامت \rightarrow برای رابطه تقدم بین دو عمل استفاده می کنیم. مثلاً $h_1 \rightarrow h_2$ به معنی آن است که عمل h_1 باید قبل از h_2 انجام شود. بازنمایی ریاضی این الگوریتم طرح ریزی در [۱۰] موجود است. در بخش بعدی کاربرد بازنمایی ماتریس حالت و الگوریتم طرح ریزی مربوطه را با یک مثال نشان خواهیم داد.

۳-۱ یک نمونه از مسئله مونتاژ

شکل (۱) یک مسئله مونتاژ را نشان می دهد. فرض کنید سیستم مونتاژ جهت جایگزینی محور قدیمی e_1 با محور جدید e_5 ساخته شده است.



شکل (۳). طرح نهایی

الگوریتم مطرح شده در یک محیط چند عامله رباتیک در دنیای بلاکها پیاده سازی شده و نتایج طرح ریزی مورد پیش بینی، به طور خودکار تولید شده است.

۴- نتیجه گیری

در این مقاله روشی جدید برای مدل سازی مسئله مونتاژ و بدنبال آن الگوریتمی معرفی شد که بطور خودکار دنباله اعمال مورد نیاز را با استفاده از بازنمایی ماتریس حالت تولید می کند و ساختار اولیه و هدف و نیز هر محدودیت عملی را مشخص می کند. از طریق این بازنمایی، با محدودیتهای مختلفی (نظیر محدودیت های منابع اولیه، محدودیتهای هندسی، ثبات، سهولت مونتاژ و غیره)، میتوانیم منطق و همراه شویم. بر مبنای این بازنمایی، تولید طرح بطور مستقیم و با عملیات ساده ماتریسی قابل پیاده سازی است. شمای بازنمایی معرفی شده و الگوریتم طرح ریزی می تواند در مسائل مونتاژ پیچیده تر و در دیگر حوزه های طرح ریزی نیز بکار برده شود.

۵- مراجع

- [1] Lavalley, .S, "planning algorithms", 2006
- [2] Vendrell, .E, Mellado, .M, Crespo, .A, "Robot planning and re-planning using decomposition, abstraction, deduction, and prediction", engineering application of artificial intelligence, pp. 505-518, 2001.
- [3] Chapman, .D, "planning for conjunctive goals", Artificial intelligence, 1987.
- [4] Currie, .K, Tate, .A, "O-plan: the open planning architecture", Artificial intelligence, 1991
- [5] Fikes, R.E, Nilson, .N, "Strips: a new approach to the application of theorem proving to problem solving", Artificial intelligence, 1971.
- [6] Rintanen, .J, Hoffmann, .J, "An overview of recent algorithms for AI planning", 2001.
- [7] Noorhosseini, S.M, Malowany A.S, "task planning for circuit board repair task in a robotic workcell", Canadian conference on Electrical and computer engineering, 1991.
- [8] Noorhosseini, S.M, Malowany A.S, "algorithmic approach to action planning for assembly automation using State matrix representation", proceedings of the 1994 IEEE international symposium on intelligent control, Columbus, OH, USA, 16-18 Aug 1994, pages 201-206.
- [9] Wilkins, D.E, "practical planning", Morgan Kaufmann, 1988.
- [10] Tsoularis, .A, Kambhampati, .C, Warwick, .k, "path planning of robots in noisy workspace using learning automata", IEEE international symposium on intelligent control, 1993.

اعمال H از طریق حل تضاد بین ماتریس غالب اعمال و S_0 مرتب می شوند:

$$F(an_{2,1})[3,1] \not\subseteq S_0 \text{ then insert } an_{3,1} \rightarrow an_{2,1} \text{ in } R.$$

$$F(an_{2,1})[4,1] \not\subseteq S_0 \text{ then insert } an_{4,1} \rightarrow an_{2,1} \text{ in } R.$$

$$F(an_{3,1})[4,1] \not\subseteq S_0 \text{ then insert } an_{4,1} \rightarrow an_{3,1} \text{ in } R.$$

$$F(an_{4,1}) \subseteq S_0;$$

$$F(an_{4,5}) \subseteq S_0;$$

اعمال H از طریق حل تضاد بین ماتریس غالب اعمال و S_* مرتب می شوند:

$$F(ap_{2,5})[4,5] \not\subseteq S_* \text{ then insert } ap_{2,5} \rightarrow ap_{4,5} \text{ in } R.$$

$$F(ap_{2,1})[3,1] \not\subseteq S_* \text{ then insert } ap_{2,1} \rightarrow ap_{3,1} \text{ in } R.$$

$$F(ap_{2,1}) \subseteq S_*;$$

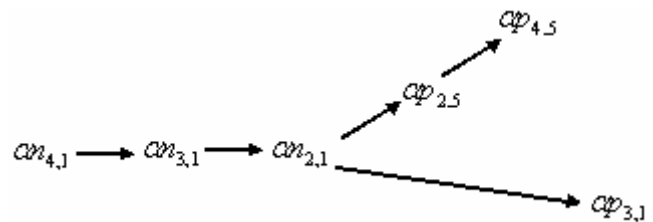
$$F(ap_{3,1}) \subseteq S_*;$$

$$F(ap_{4,1}) \subseteq S_*;$$

حال مجموعه R بصورت زیر خواهد بود:

$$R = \left\{ \begin{array}{l} an_{2,1} \rightarrow ap_{2,5}, an_{3,1} \rightarrow ap_{3,1}, an_{4,1} \rightarrow ap_{4,5} \\ an_{3,1} \rightarrow an_{2,1}, an_{4,1} \rightarrow an_{2,1}, an_{4,1} \rightarrow an_{3,1} \\ ap_{2,5} \rightarrow ap_{4,5}, ap_{2,1} \rightarrow ap_{3,1} \end{array} \right\}$$

سپس شبکه طرح تولیدی برای وجود دور بررسی می شود. از آنجا که هیچ دوری در شبکه وجود ندارد، طرح با شکست مواجه نخواهد شد. سپس شبکه طرح هرس شده و در یک مجموعه طرح P ذخیره می شود. شکل (۲) طرح تولید شده در این مرحله را نشان می دهد.



شکل (۲). شبکه اعمال هرس شده.

گام ۳: ترکیب اعمال. در این مرحله آن اعمالی که می توانند با یکدیگر ترکیب شوند پیدا شده و بوسیله اعمال پیچیده جایگزین میشوند. ترتیبی نظیر $an_{2,1} \rightarrow ap_{2,5}$ در p وجود دارد که می توانند با یکدیگر ترکیب شوند. از اینرو، آن را با عمل ترکیبی $\alpha_{2,1,5}$ جایگزین می کنیم. شکل (۳) طرح نهایی را نشان می دهد.