

Приложение 2

Анализ на методи за генериране на конфигурационно пространство на работи.
Класификация на алгоритми и подходи.

Приложение към научен отчет на проект:

№ КП-06-Н87/5 от 06.12.2024 г.

със заглавие „Изследване на работи и алгоритми за работа в сложна среда „
Финансиран от ФНИ по „Конкурс за финансиране на фундаментални научни изследвания –
2024 г.“

София, 2026г.

Съдържание

1. Методи за генериране на конфигурационно пространство на работи	3
1.1. Основни понятия	3
1.2. Права и обратна задача на кинематиката в роботиката	4
1.3. Подходи за решаването на обратната задача на кинематиката (ИК). Анализ, предимства и недостатъци	5
2. Методи за генериране на конфигурационно пространство при мобилни работи	7
3. Методи за генериране на конфигурационно пространство при стационарни работи	9
4. Анализ на съществуващите методи за генериране на работно и конфигурационно пространство при стационарни и мобилни работи	11
5. Класификация на алгоритми и подходи за планиране на движения в сложна среда	13
6. Заключение	19
Литература	19

1. Методи за генериране на конфигурационно пространство на работи

За да се постигне пълнота и яснота на изложението в това приложение най-напред се поясняват някои основни понятия и определения. След това се разглеждат и систематизират различните известни методи за решаване на обратната задача на кинематиката, които са в основата за намиране на конфигурационното пространство и по голямата част от алгоритмите за планиране на движения в среда с препятствия.

1.1 Основни понятия

Конфигурация на робот - Всяко едно възможно взаимно разположение на звената на робота се нарича конфигурация. Конфигурацията обикновено се означава с q . По време на изпълнение на задачите роботът постоянно променя своята конфигурация. Някои по-важни конфигурации са:

- *“Home position”* – начална позиция. Тази конфигурация е избрана за стартова при изпълнение на задачите от робота. Позицията може да се определи еднозначно от сензорите на робота. Ако е възможно често се преминава през нея, за да се повиши точността на изпълнение на задачите.
- *Крайни позиции* – Звената на робота обикновено не могат да се преместват и завъртат безкрайно. Тези ограничения настъпват поради конструктивни съображения или са наложени от системата за управление. Наричат се ставни ограничения. Когато едно или повече звена на робота са в такава крайна конфигурация се намаляват двигателните способности на робота в определена посока. Крайните позиции обикновено се избягват при програмиране на робота.
- *Сингулярни конфигурации* – това са особени конфигурации, при които обикновено намаляват степените на свобода на робота. Кинематиката на работи ги разглежда по-подробно. В повечето случаи са нежелателни.

Тези понятия са ясни когато се прилагат към стационарни работи, но когато се интересуваме от мобилни работи може да се отбележат някои разлики и особености. Когато разглеждаме мобилни работи обикновено се интересуваме от позицията и ориентацията му спрямо заобикалящата го среда, което се приема за конфигурация. Все пак съществува голямо многообразие от мобилни работи, затова трябва да се съобразим с конкретните условия. При крачещите работи конфигурацията ще се определя от състоянието на краката спрямо тялото на робота, освен позицията и ориентацията му като цяло. Когато се планират движения на хуманоиден робот понятието конфигурация ще включва още повече елементи.

Конфигурационно пространство – множеството от всички конфигурации $q(i)$, които може да заема робота. Обикновено се означава с C . Това е основно понятие в роботиката и

планирането на движения. Конфигурационното пространство представя едно състояние на робота като точка в него. Това е много удобно, защото позволява лесно да се прилагат различни математически и алгоритмични методи за изследване на поведението на точка в n -мерно пространство.

1.2. Права и обратна задача на кинематиката в роботиката

Правата задача може да се дефинира така: *Да се определят позицията и ориентацията на изпълнителното звено на робот по зададени обобщени координати (ставните ъгли) при известни геометрични параметри на робота.* Правата задача обикновено дава отговор и на въпроса за взаимното разположение на звената на робота едно спрямо друго. В някои случаи правата задача се разглежда на ниво скорост, ускорение или сила. Тогава се формулира така: *Да се определят линейните и ъгови скорости (ускорения/сили) на изпълнителното звено на конкретен робот при известни скорости (ускорения/ сили) на двигателите в ставите му.* С други думи изследва се връзката между движенията на ставите и движенията на изпълнителното звено.

Позицията на изпълнителното звено и ориентацията му в пространството зависи от конфигурацията q на ставните ъгли. Математически това може да бъде изразено като функционална връзка между вектора на позицията и ориентацията на хващача $x = [x_p \ y_p \ z_p \ \alpha_x \ \alpha_y \ \alpha_z]^T$, и вектора на конфигурацията $q = [\theta_1 \ \theta_2 \ \theta_3 \dots]^T$ като:

$$x = f(q). \quad (1)$$

Тази връзка е известна като права кинематика (*Forward kinematics -FK или Direct Kinematics - DK*).

Кинематичните (*Inverse Kinematics - IK*) уравнения се използват в роботиката, компютърните игри и анимацията. Предлагат се различни подходи за решение на проблема, като всеки един от тях си има своите предимства и недостатъци. Търсенето на решение в голяма степен зависи от топологията на механизмите.

Обратната кинематика е един от най-важните, най-изучаваните и най-трудни проблеми в роботиката, тя има за цел да изчисли стойностите на променливите на ставите, като се има предвид желаната позиция и ориентация на крайния ефектор на робота. Очевидно резултата от обратната кинематика е в основата на управлението на роботите.

Кинематиката на роботите се занимава с два вида проблеми - проблеми с директна и обратна кинематика. Серийните роботи са най-разпространените индустриални манипулатори. Те са проектирани като поредица от звена, свързани чрез моторно задвижвани стави, които се простират от основа до краен ефектор. За серийните роботи проблемът с директната кинематика е ясен и лесно се намира едно-единствено решение; най-популярният метод за решението е предложен от Денавит и Хартенберг [1].

1.3. Подходи за решаването на обратната задача на кинематиката (ИК). Анализ, предимства и недостатъци

Обратната кинематика (ИК) е много по-труден проблем от директната кинематика поради наличието на сингулярности и нелинейности. Напълно аналитични решения съществуват само за малък клас кинематично прости манипулатори [2]. За много конструкции на индустриални роботи са намерени специфични обратни кинематични решения, използващи различни алгоритмични подходи, които не включват директни аналитични решения. Извеждането на кинематични модели за роботи със серийна топология е от съществено значение за анализа, изследването и програмирането на индустриални роботи.

- Аналитичен подход за решение на ИК

В [2] е описано аналитично решение за робот с две звена, както и за основната структура на Станфордския манипулатор. Изследването в [3] изследва популярни механизми със серийна структура с две и три степени на свобода, чиято обратна кинематика позволява аналитични решения в затворена форма. За механизмите със серийна структура е разработена програма на Python.

- Геометричен подход за решение на ИК

Геометрични решения на задачата с обратната кинематика за роботи с три и шест ротационни степени на свобода са дадени съответно в [4] и [5]. Геометричният подход обикновено е по-интуитивен и значително намалява сложността и времето за изчисления на ИК. Той обаче изисква индивидуален подход към всеки различен тип манипулатор.

- Хибридни методи и алгоритми

В [6] е предложен хибриден алгоритъм, който комбинира геометричния метод с аналитичния и е приложен към индустриален робот PUMA 560. В [7] обратната кинематична задача е решена с редуциран набор от уравнения, подходящ избор на координатната система, свързана с изпълнителното звено, и прилагане на принципа на ортогоналност при въртене. Подходът не изисква изчисляване на задачата за права кинематика и може да се използва за манипулатори с различна геометрия.

- Особености при редундантните роботи

За редундантни роботи, аналитичното определяне на всички решения на обратната кинематична задача е много трудна задача [8,9,10,11]. Въпреки сложността, [8,9,10,11] съдържат няколко примера за аналитично намиране на решения. Роботизираната ръка Pioneer 2 е редундантен робот с пет степени на свобода. Аналитичните уравнения за неговата обратна кинематика са изведени в [8]. Статия [9] представя решения на правата и обратните задачи на редундантен робот Power Cube 4-DOF, използвайки параметрите на Денавит и Хартенберг, косинусовата теорема и метода на векторните координати. Показано е, че различната позиция и промяната на допълнителна входна променлива имат значително влияние върху конфигурацията на робота. Човешката ръка е моделирана като редундантен сериен манипулатор в [10]. Определено е нулевото пространство на матрицата на Якоби за предложения модел. Резултатите в [10] показват, че излишъкът в човешката ръка се използва за намаляване на грешките в траекторията и анизотропията, произтичащи от външни смущения. Авторите на [11] предлагат метод за ускоряване на изчисленията в аналитичните

и числените методи за решаване на обратната кинематична задача за робот със седем степени на свобода.

- Числени методи

Числените алгоритми обикновено не дават всички решения. Като цяло, изборът на начална стойност и алгоритъмът за търсене имат голямо влияние върху точността на тяхното решение [6]. В [12] е предложен числен метод за инверсна диференциална кинематика на серийни роботи, който използва двойна кватернионна формулировка на кинематика в алгоритъм със сходящи пътища. Според [12], традиционните алгебрични, геометрични и итеративни методи са неадекватни, ако структурата на манипулатора е по-сложна. С увеличаване на сложността на робота, решението на обратната кинематична задача изисква повече изчислително време. В [13] е използвана способността на ANFIS (Адаптивна Невро-Размита Система за Извод) да се учи от данни за обучение за решаване на обратната кинематична задача. Проведени са компютърни симулации за роботи с две и три степени на свобода, които демонстрират подхода. Числените методи, базирани на обратна диференциална кинематика, са ефективни при решаване на обратната кинематична задача на роботи със седем степени на свобода с произволни геометрични параметри. Те обаче предлагат недостатъчна числена стабилност и отнемат време за определяне на уникално решение. Изследването в [14] представя подход, който генерира общ алгоритъм за n-звенов планарен хиперредундантен робот. Този метод многократно използва решението на обратната задача на двузвенов робот върху виртуални стави, където виртуалните стави се дефинират с помощта на геометрично представяне.

- Неврони мрежи

В [15] се използват локално рекурентни невронни мрежи (LRNN) за решаване на обратната кинематична задача. Представена е MATLAB/Simulink симулация за манипулатор с шест степени на свобода чрез изчисляване на обратната кинематика с LRNN и прилагане на сложни движения. Авторите на [16] представят анализ на производителността на различни подходи за решаване на обратна кинематика с невронни мрежи и резултатите са сравнени с аналитичния подход. Основната причина за използването на техники, базирани на данни, като невронни мрежи, за решаване на обратната кинематика на роботизирани манипулатори е, че тя може да бъде разширена до произволен брой стави без много усилия, докато други методи трябва да вземат предвид броя на ставите и видовете връзки предварително. В [17] са създадени набори от данни за обучение на невронни мрежи със симулационен софтуер, който използва кинематичен модел на робота. Решението на всяка невронна мрежа се оценява с помощта на аналитично решение на проблема с директната кинематика на робота.

- Метод на координатно спускане

Цикличното координатно спускане (CCD) е числен метод, който използва само геометрични съображения за извеждане на последващи итерации. Идеята е всяка връзка да се движи независимо, така че да се минимизира грешката между крайния ефектор и целта, като се повтаря това върху всички стави (повтаряйки процеса итеративно) [18,19]. Това е популярен метод за решаване на обратна кинематика, който намира приложение в компютърната анимация на герои [18,20] и в роботиката [19]. В [19] този метод е комбиниран с метода на Нютон и е приложен с шест различни известни роботизирани манипулатора. CCD алгоритъмът може лесно да се приложи в роботиката, но може да отнеме серия от итерации, преди да се достигне решение, а също така може да генерира неправилни

завъртания на ставите. Констатациите в [20] представят алгоритъм, който използва целеви триъгълник, за да определи ориентацията и ъглите на ставите, като елиминира проблемите, свързани с неправилни и големи ъглови завъртания. Авторите на [21] предлагат метод, използващ косинусовата теорема, който бързо определя ъглите на ставите на кинематична структура, когато целта е дадена. Той комбинира аналитичен с числен метод.

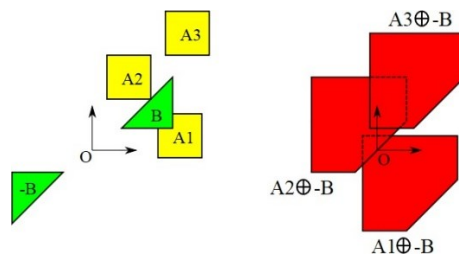
Изследването в [22] предлага модел на структурирана изкуствена невронна мрежа (ANN) и адаптивна система за размита невронна инференция (ANFIS), за да се намери обратното кинематично решение на робот. Тъй като няма уникално обратно кинематично решение, към началната конфигурация се прилагат предсказващи модели. Като цяло, ИК се влияе от неправилна конфигурация на връзките, което означава, че съществуването, уникалността и стабилността на неговите решения не са гарантирани [23]. В [23] авторите демонстрират ефективността на прилагането на мрежова инверсия с регуларизация, чрез което може да се намали неправилното позициониране. Даден е пример с роботизирана ръка с множество стави.

2. Методи за генериране на конфигурационно пространство при мобилни работи

Алгоритмите за планиране на движение на базата на сума на Минковски обикновено включват изчисляване на конфигурационни пространства (C- пространства), въведени от Lozano- P´erez (1983) [24] за планиране на движение на обект сред физически препятствия. Всяка точка в C-пространството съответства на набор от параметри, които характеризират позицията и ориентацията на твърдия обект. Съществуват някои съществени различия при определяне на конфигурационното пространство при мобилни и стационарни работи. Първо ще разгледаме конфигурационното пространство при мобилни работи, което обикновено се определя по-лесно. За нашите цели е достатъчно да знаем, че препятствията в работното пространство C , го разделят на две групи, *свободното пространство* C_{free} и *пространство, заето от препятствия* C_{obs} , където $C = C_{free} \cup C_{obs}$.

C_{obs} е наборът от всички конфигурации на робота, за които той ще се сблъска с едно или повече от препятствията, докато свободното пространство C_{free} е набора от конфигурации, при които той не се сблъсква. След това проблемът с планирането на движението се преразглежда, така че да изчислим C_{obs} и C_{free} и да намерим път в свободното пространство - свързване на началната и целевата конфигурации.

Изчисляването на C-пространствата при мобилни работи се основава на сумите на Минковски: ако A и B са две множества от точки, които се пресичат помежду си, тогава началото на координатната система O трябва да лежи в множеството получено от сумата $A \oplus - B$. Нека разгледаме прост пример с робот B с триъгълна форма. Приемаме фиксирана ориентация на мобилния робот, така че конфигурацията се определя само от позицията. Фигура 1 показва 2D илюстрация. Обърнете внимание, че началото O може да бъде избрано произволно, тъй като промяната на началото O е еквивалентно на прилагането на еднаква трансляция към двете групи A и B , така че не се променя нито взаимната позицията, нито формата на сумата $A \oplus - B$.



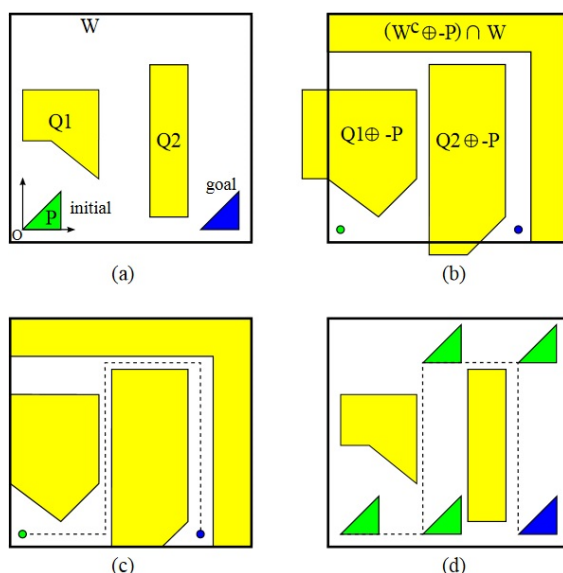
Фиг. 1. Откриване на сблъсък чрез използване на суми на Минковски. Препятствието A_1 се пресича с обекта B , препятствието A_2 просто докосва B на границата и A_3 е далече от B .

Пространствата C_{obs} и C_{free} обикновено се изчисляват, като се използват суми на Минковски. За движещ се обект P и обединението на всички препятствия Q (виж Фигура 2, $Q = Q_1 \cup Q_2$), в пространство заето от препятствия C_{obs} е $Q \oplus -P$. В пространството на конфигурациите препятствията се увеличават с помощта на сумата на Минковски и движещият се обект се свежда до една точка. По този начин проблемът за откриване на сблъсък между 3D обекти се свежда до по-простият проблем за откриване на сблъсък между една точка и набор от 3D обекти. Свободното пространство C_{free} е разликата от цялото достъпно от обекта пространство C и пространство, заето от препятствия C_{obs} . Може да се напише:

$$C_{obs} = Q \oplus -P \quad (2)$$

$$C_{free} = (Q \oplus -P)^c \quad (2)$$

Обикновено движещият се обект е ограничен в работно пространство, обозначено като W . Може да си мислим за W като за стени на стая, в която се движи мобилен робот. За да вземете под внимание стените на работно пространство, може да разгледате допълнението на работното пространство (W^c) като препятствие, така че обектът да не се „докосва“ т.е. да се сблъска с външната страна на работното пространство.



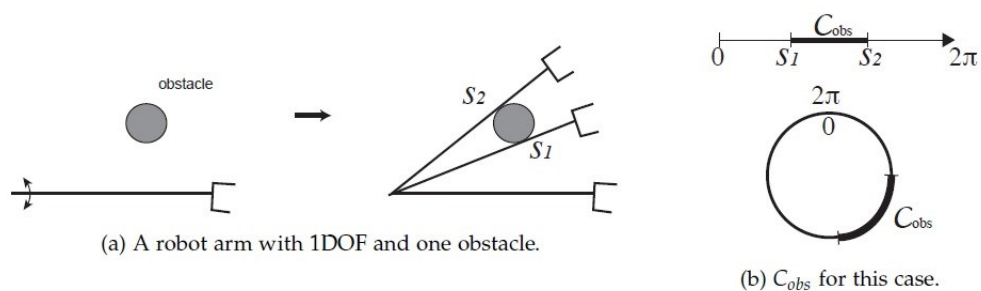
Фиг. 2. Планиране на движение на базата на C -пространство. а) движещ се обект (робот) P и две препятствия Q_1 и Q_2 в работно пространство W . Искаме да преместим P от първоначалната в целевата позиция. (b) C -пространство, тук препятствията Q_1 и Q_2 са увеличени с размерите на робота P , като се ползва сума на Минковски. (c) Път, съдържащ

се в свободното пространство, свързващо начална и целева конфигурация. (d) Движението на обекта в работното пространство.

3. Методи за генериране на конфигурационно пространство при стационарни работи

Определянето на това дали робот при конфигурация q е в сблъсък с предмет обикновено е сложна операция, включваща CAD модел на околната среда и робота. Ставните ограничения също могат да се разглеждат като препятствия в конфигурационното пространство. Когато обаче има поставено препятствие в работната зона на робот ситуацията е по-сложна. Ако разгледаме гранично прост случай на стационарен „робот“ с една ротация и препятствие ще открием част от проблемите (фиг.3).

For an arm with one degree of freedom, C_{obs} looks like as:



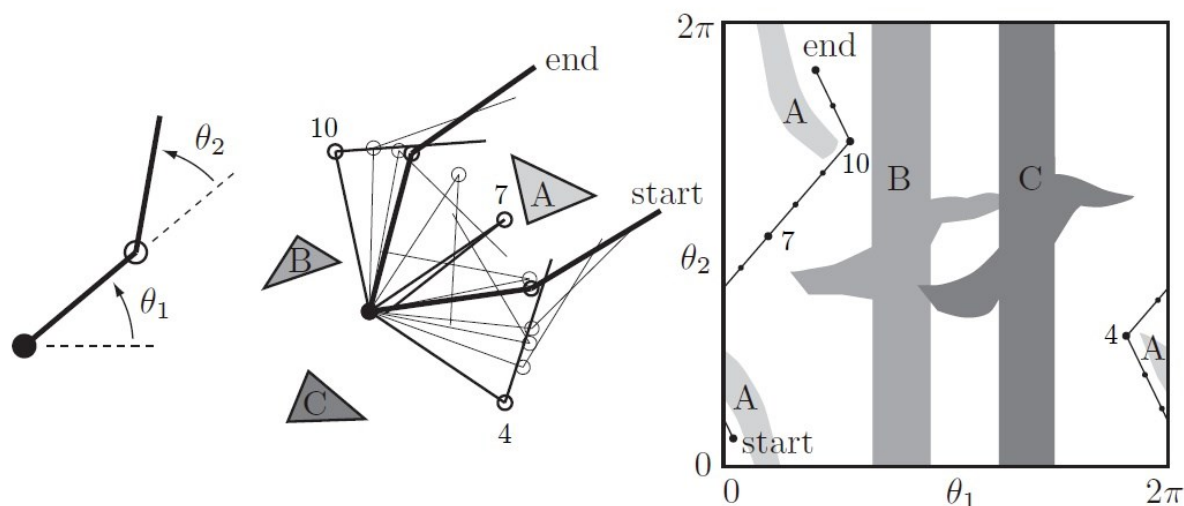
Фиг.3. Ротиращо звено, препятствие и представянето му в едномерно конфигурационно пространство.

За разлика от генерирането на конфигурационно пространство C при мобилни работи, тук едно и също препятствие (и един и същ робот) ако му променим позицията ще генерира различни (C_{free} и C_{obs}) свободни и заети конфигурационни пространства. Освен това ставните ограничения оказват съществено влияние върху процеса на определяне на C . Всяка точка в C -пространството съответства на уникална конфигурация q на стационарния робот и всяка конфигурация на робота може да бъде представена като точка в C -пространство. Например, конфигурацията на робот с n стави може да бъде представена като списък от n позиции на ставите, $q = (1, 2, \dots, n)$. Свободното C -пространство - C_{free} се състои от конфигурации, при които роботът не влиза в конфликт с препятствие, нито нарушава допустимите граници. Приема се, че q е n -мерен вектор и $C \subset \mathbb{R}^n$ (е подмножество на n – мерното пространство). Управляващите входове (сигнали) на робота са написани като m -мерен вектор: $u \in U \subset \mathbb{R}^m$, където $m = n$ при типичните работи с пълноценна кинематика.

С концепциите за C_{free} и C_{obs} , проблемът за планиране на пътя се свежда до проблем за създаване на път на „точков робот“ сред препятствията C_{obs} . Ако препятствията прекъсват C_{free} в отделни свързани участъци, като q_{start} и q_{goal} не лежат в една и съща свързана област, тогава няма път без сблъсък. Достоверното математическо представяне на пространството, заето от препятствията може да бъде изключително сложно и поради тази причина то рядко се представя точно. Въпреки това, концепцията за C_{obs} (Пространство на препятствията) е много важна за разбирането на алгоритмите за планиране на движението. Идеите се илюстрират най-добре с примери.

Пример. RR равнинен робот. На фиг.4 е показан RR равнинен робот, като смятаме, че няма наложени ставни ограничения. Една конфигурация на робота се представя с точка

$q(\theta_1, \theta_2)$. В работното пространство са поставени триъгълни препятствия А, В и С. Конфигурационното пространство - С на робота е представено като част от равнината $0 \leq \theta_1 < 2\pi$, $0 \leq \theta_2 < 2\pi$. В този случай, топологията на С-пространството е торус, тъй като ръбът при $\theta_1 = 2\pi$ е свързан с ръба при $\theta_1 = 0$; аналогично, при $\theta_2 = 2\pi$ е свързано с $\theta_2 = 0$, защото няма наложени ставни ограничения. Квадратната област в R^2 (равнината) се получава чрез разрязване на повърхността на торуида два пъти, при $\theta_1 = 0$ и $\theta_2 = 0$, и разпъване в равнината.



Фиг.4. (Ляво) Ставни ъгли на RR робот. Ръката се движи сред препятствия А, В и С. (Дясно) Същото движение в С-пространството. Обозначени са три междинни точки, 4, 7 и 10 от пътя.

С-пространството отдясно на фигура 4 показва препятствията А, В и С поставени в работното пространство, представени като С-препятствия в конфигурационното пространство (пространство на ставните ъгли). Всяка конфигурация, намираща се в C_{obs} , съответства на сблъсък на звено от робота с препятствие. В работното пространство и в С-пространството е показан свободен път за ръката на робота от една конфигурация до друга. Пътят и препятствията илюстрират топологията на С-пространството. Обърнете внимание, че препятствията прекъсват C_{free} на три свързани компонента.

За да се намери конфигурационното пространство (съответно C_{obs} и C_{free}) за стационарен робот трябва многократно да се търси решение на обратната задача на кинематиката (ИК). Основните методи за това се обясниха в точка 1.3. Друг важен елемент в този процес е използването на алгоритми за откриване на колизии между обектите. Конфигурационното пространство в общия случай е многомерно, което допълнително усложнява задачата за търсене на път без сблъсък в него. Поради тези причини се търсят методи за опростено представяне на С-Прасе, което в повечето случаи не е пълно но е достатъчно за да се планира движение без сблъсък.

4. Анализ на съществуващите методи за генериране на работно и конфигурационно пространство при стационарни и мобилни роботи.

Методи базирани на клетъчно разделение. Те ползват разделяне на пространството на клетки или мрежа от точки, за част от които (които не са в препятствие) се решава ИК и се получават съответни точки в конфигурационното пространство. Тези методи са подходящи за пространства с малко измерения (роботи с малък брой степени на свобода).

Предимствата им са свързани с:

- Простота за представяне на обектите;
- Лесно се описват алгоритмично;
- Подходящи са за приложение към мобилни роботи;
- Прилагат се при прости по-форма препятствия с малки разлики в размерите.

Главните им недостатъци са:

- При по-сложни препятствия е необходимо да се увеличи значително броя на клетките. Това увеличава изчислителното време;
- При повече от две измерения се прилага трудно.

Един от начините да се намали изчислителната сложност на алгоритмите, базирани на мрежа, е да се използва решетка с различна разделителна способност за C_{free} . За да бъде представено коректно препятствието, клетката с препятствия се разделя на по-малки клетки. Всяко измерение на оригиналната клетка се разделя на половина, което води до 2^n под-клетки за n -мерното пространство. Всички клетки, които все още са в контакт с препятствие от конфигурационното пространство C , се разделят допълнително до зададена максимална резолюция. Всички мрежови методи, осигуряват оптимални решения, подчинени на избраната дискретизация. Недостатък на тези подходи обаче е тяхната висока изчислителна сложност, което ги прави неподходящи за системи, които имат повече от няколко степени на свобода.

Различен клас алгоритми, известни като методи за вземане на проби, разчитат на произволна или детерминирана функция за избиране на извадка точки от конфигурационното C пространство [25]. Те съдържат:

- *функция за оценка* дали дадена проба или движение е в C_{free} ;
- *функция за определяне на подходящи предишни проби* от свободното пространство;
- и *местен алгоритъм*, който да се опита да се свърже с новата проба или да се придвижи към нея.

Тези функции се използват за изграждане на граф или граф дърво, представляващи възможните движения на робота.

Методите за вземане на проби обикновено се отказват от оптималните решения за търсене в мрежа в замяна на възможността бързо да намерят задоволителни решения в пространства на състоянието с голям брой измерения. Пробите се избират, за да формират пътна карта или дърво за търсене, което бързо се разпространява в свободното пространство C_{free} , използвайки по-малко образци, отколкото обикновено се изисква от фиксирана мрежа с висока резолюция, където броят на точките на мрежата нараства експоненциално с размерността на пространството за търсене. Повечето методи за вземане на проби са

вероятностно пълни: това означава, че вероятността за намиране на решение, когато такова съществува, се доближава до 100%, когато броят на пробите клони към безкрайност.

Двата основни класа на извадковите методи са: бързо проучване на случайни дървета (RRT - rapidly exploring random trees) и вероятностни пътни карти (PRMs - probabilistic roadmaps). Първите използват дърво за представяне и планиране с една заявка или в C-пространството, или в пространството на състоянието, докато PRM са предимно алгоритми за конфигурационното C-пространство, които създават пътната карта за планиране на няколко заявки.

Предимствата им са:

- Сравнително висока скорост. Търсят решения само за определен брой точки. Това позволява много бързо да се открие път в среда с малко и прости препятствия;
- Може да се прилага за работи с повече степени на свобода;
- Точките обикновено се генерират в конфигурационното пространство, след което с правата задача се трансформират в реалното пространство и накрая се проверява за колизия. Правата задача на кинематиката се решава по-лесно от обратната при серийни работи, които се ползват най-често за работа в среда с препятствия.

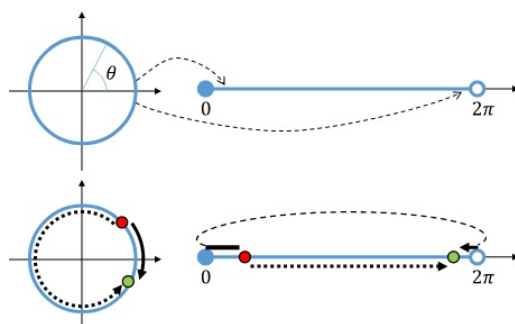
Главните им недостатъци са:

- Не намират най-краткия път между началната и крайната конфигурация.
- При сложни препятствия и голям брой изчислителното време нараства.

Изпълнителното звено на робота трябва да достигне до различни точки от работното пространство с определени ориентации. Ротациите обаче се държат по коренно различен начин от преместванията. За да разберем по-добре ротациите, първо ще представим малко елементи от топологията, клон на математиката, който изучава как са свързани пространствата. Свойствата на топологичната свързаност се запазват при произволни обратими трансформации и следователно те са основни характеристики на самите пространства, а не просто техните координатни представяния. Ще разгледаме топологични понятия от гледна точка на пространството на 2D ротации $SO(2)$. Непосредствен проблем, на ротациите в равнината е, че ъглите се „обгръщат“, така че посоката при 0° е идентична с тази при 360° . Това е важен въпрос в много аспекти:

- За да се върти звено на робот от начален ъгъл 30° до краен ъгъл 330° е по-бързо да завъртите на -60° , а не $+300^\circ$.
- Ако ставата на робота има по-голям от 360° диапазон на движение, тогава нейното вътрешно положение в даден момент може да бъде измерено само външно ($\pm k360^\circ$), т.е. за да се оцени абсолютното положение с помощта на външни средства, трябва да се запомни историята на всички движения на ставата.
- Когато извлича елементи от таблица (например на контролни команди или движения), индексирани с ъгъл, човек трябва да внимава да извлече най-близкия елемент по отношение на абсолютното ъглово отклонение, вместо просто да приемаме абсолютната разлика на ъглите.

Въпреки че между всеки елемент на $SO(2)$ и ъгъл в интервала $[0, 2\pi)$ има индивидуално съпоставяне, въпросът за „обхождане“ илюстрира, че $SO(2)$ по същество не е еквивалентен, тъй като неговите елементи не се държат като числа в интервала, когато се обсъжда непрекъснатост и близост, както е показано на фиг.5.



Фиг.5. Отгоре: Топологията на пространството на 2D ротации е хомеоморфна на окръжност, но не и на линеен интервал. Въпреки че между точките на окръжността и полуотворения интервал $[0, 2\pi)$ има индивидуално съпоставяне, прекъсването при 2π ги прави топологично различни. Долу: Най-краткият път между двете маркирани точки на окръжността (плътна стрелка) обхваща област около 2π , което е прекъснат път на интервала. Най-краткият непрекъснат път на интервала (дебелата пунктирна стрелка) води до по-дълъг път по окръжността.

5. Класификация на алгоритми и подходи за планиране на движения в сложна среда

Класификация според вида на средата. Представянето на средата може да е предварително известно или може да бъде открито чрез методи за локализация и картографиране, при планирането на пътя на робота. В зависимост от вида на препятствията проблемите с планирането на пътя могат да бъдат класифицирани в две категории.

- 1) Проблеми с планирането на пътя в статична среда.
- 2) Проблеми с планирането на пътя в динамична среда.

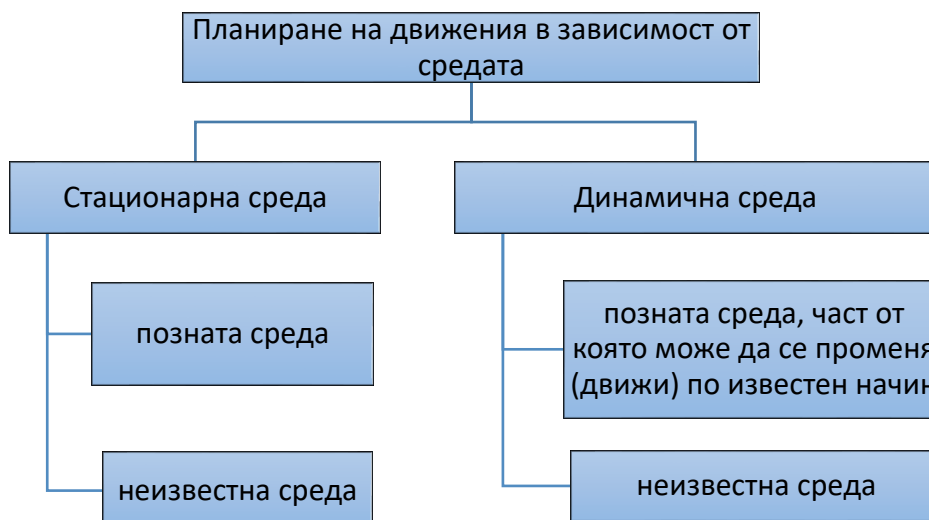
При проблемите с планирането на траекторията в статична среда, цялата информация за околната среда е известна предварително и е неподвижна. Този тип проблеми с планирането на пътя се наричат проблеми с планирането на пътя в известна среда. След като пътят е планиран преди роботът да започне да се движи, няма да има промени в определения път по време на движението на робота.

В случай на планиране в динамична среда (част от околните препятствия са подвижни). Често имаме само частична информация за околната среда и поведението на обектите в нея, (обикновено видимите части от препятствията). Когато роботът достигне дадена междинна цел, той планира пътя въз основа на наличната информация. Докато роботът следва пътя си, той открива повече информация за препятствията и това се използва за актуализиране на информацията за околната среда и репланиране на пътя. Този процес продължава, докато роботът не достигне целта си.

Всяка от горните две категории (1 и 2) може да бъде допълнително разделена на две подкатегории въз основа на въпроса доколко „роботът знае“ цялостната информация за околната среда.

- Планиране на пътя на робота в ясно позната среда, където роботът вече знае картата на околната среда, преди да започне да се движи.

- Планиране на пътя на робота в частично известна или несигурна среда, където роботът няма познания за околната среда.



Фиг.6. Класификация на проблема за планиране на движения в зависимост от средата

1) Планиране на пътя в статична известна среда. Роботът знае цялата информация на околната среда, преди да започне да се движи. Следователно роботът може да изчисли оптималните движения без сблъсък в свободния път офлайн преди мисията.

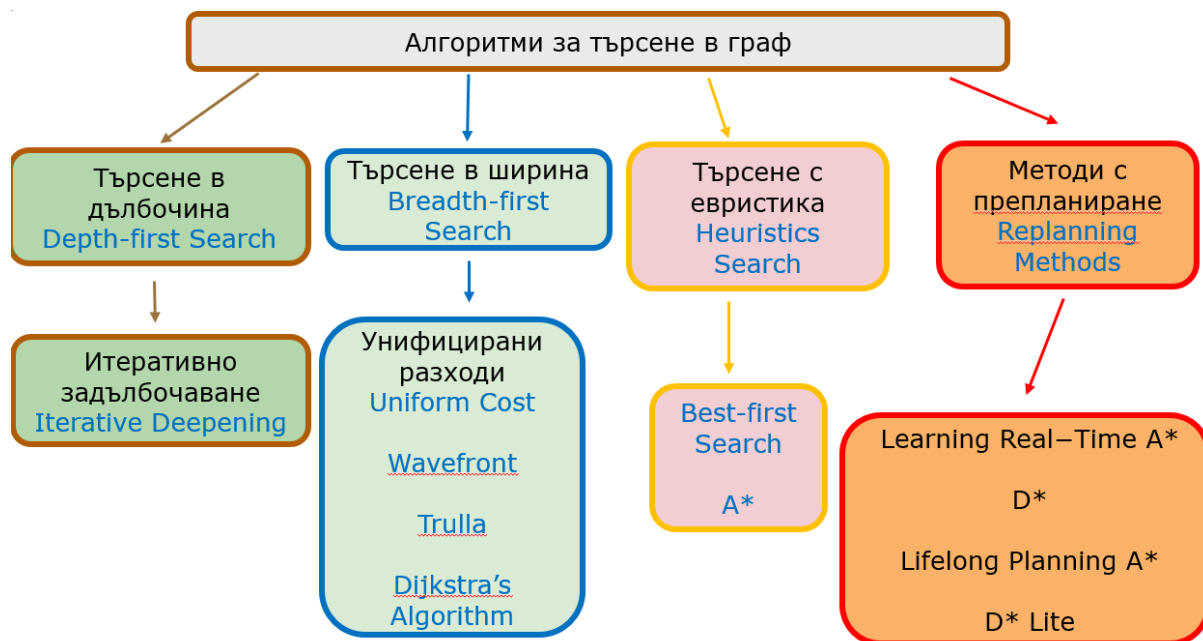
2) Планиране на пътя в динамична известна среда. В този вид среда роботът има знания за другите движещи се обекти, това означава, че роботът е способен да получава информация за тяхното местоположение, скорост, ускорение и др. Такава информация може да бъде получена от всеки движещ се обект или от централна комуникационна система.

3) Планиране на пътя в непозната статична среда. В статична и неизвестна среда навигацията на робота е по-трудна от тази в статична и позната среда. Това се дължи на несигурността на информацията, получена от сензорите за околната среда. Следователно оптималният резултат не може да бъде получен предварително и роботът трябва да използва локална информация за изчисляване на пътя, докато се движи.

4) Планиране на пътя в динамична неизвестна среда. Проблемите с планирането на пътя в динамична и неизвестна среда е най-сложния случай в планирането на пътя на робота. Оказва се, че това е най-често срещаната ситуация, с която робота трябва да се справи. В такава динамична неизвестна среда, роботът не може да използва никаква предварителна техника за планиране на глобален път, следователно планирането на локални пътеки е най-добрият подход. Невъзможно е да се получи глобална оптимизация. Роботът трябва да бъде снабден с много сензори за получаване на информация за околната среда. Направете онлайн планиране на пътя в реално време. Времето за планиране на робота трябва да е много кратко, защото роботът има малък интервал от време, да коригира движението си, за да се избегнат движещите се препятствия.

Много алгоритми за движение пряко или косвено представят С-пространството или пространството на състоянието като граф. Графът се разглежда като съвкупност от върхове (възли) и дъги (ребра). Използва се за представяне на съвкупност от обекти и техните връзки. Обикновено върховете съответстват на обектите, а дъгите — на отношенията между тях.

Графът е структура от данни, която се използва за решаването на редица интересни задачи от практиката. Много задачи от различни области на науката и практиката могат да бъдат моделирани с граф и решени, като се изпълни съответният алгоритъм върху него. Графът и дърветата могат да бъдат представени по-компактно като списък с възли, всеки с връзки към неговите съседи. На фиг.7 са представени основните алгоритми за търсене в граф, които се прилагат в роботиката.



Фиг.7. Основни алгоритми за търсене в граф, които намират приложение при планиране на движения в среда с препятствия.

Методи на пълното планиране

Пълното планиране на пътя разчита на точното представяне на свободното пространство C_{free} . Метода дава много добри резултати при прости системи с две степени на свобода и малък брой препятствия с известна форма. Тези техники обаче са склонни да бъдат математически и алгоритмично сложни и непрактични за много реални системи. Един подход за пълно планиране на пътя, се основава на представянето на сложното многомерно пространство C с *едномерна пътна карта* R със следните свойства:

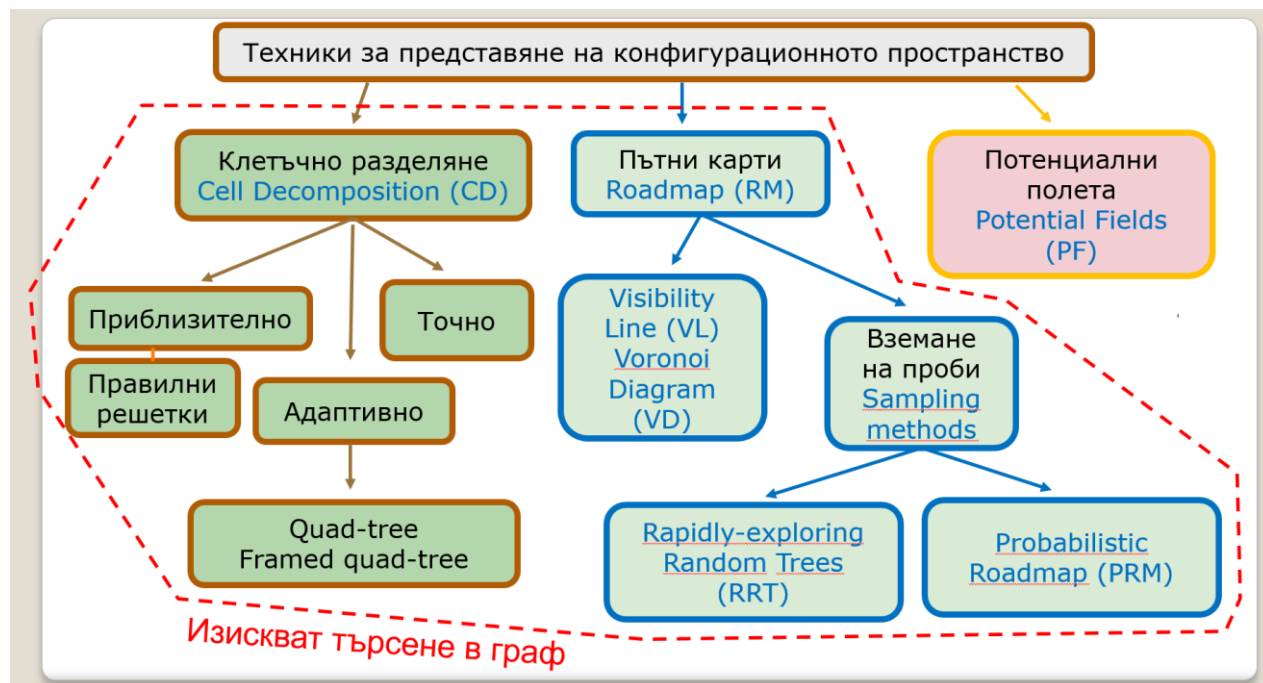
а) Достъпност. От всяка точка $q \in C_{free}$ на свободното конфигурационно пространство, свободен път до точка $q' \in R$ от пътната карта може да бъде намерен тривиално (напр. Линеен път).

б) Свързване. За всеки свързан компонент (област) от C_{free} има свързани компоненти от пътната карта R . С други думи в свободното пространство могат да се свържат точки, които да образуват пътна карта, по която да се движи робота (обекта).

Пътната карта включва множество възли и връзки между тях, но няма определена целева и стартова точка. С такава пътна карта алгоритъмът може да намери път между две точки q_{start} и q_{goal} в същата свързана област на C_{free} , чрез просто търсене на път от стартовата точка q_{start} до точка $q'_{start} \in R$, от точка $q'_{goal} \in R$ до $q_{goal} \in R$, и от q'_{start} до q'_{goal} на пътната карта R . Ако пътят може да бъде намерен тривиално между q_{start} и

q_{goal} , пътната карта може дори да не се използва (директно свързване на стартовата и целевата точки).

Поради сложността свързана с точното представяне на обектите тези методи са намерили ограничено приложение. На фиг. 8 са дадени основните методи за планиране на път, които използват представяне на информацията в конфигурационното пространство.



Фиг.8. Алгоритми за търсене на път без сблъсък, които обикновено са свързани с представяне на граф в конфигурационното пространство.

Планиране на пътя с граф на видимост (Visibility graph)

В литературата може да го срещнете под името линия на видимост (Visibility Line (VL)). Този метод изчислява най-късите Евклидови пътища (Euclidean shortest paths), които може да измине мобилен робот в среда с многоъгълни препятствия. За първи път метода е ползван в проекта Shakey (Visibility graph - Shakey Project, SRI [Nilsson, 1969], https://en.wikipedia.org/wiki/Shakey_the_robot). Проектът съчетава изследвания в роботиката, компютърното зрение и обработката на естествен език. Поради това е първият проект, който съчетава логически разсъждения и физически действия. Shakey е разработен в центъра за изкуствен интелект в Станфорд. В основата на метода стои проста идея да се движим по прави линии към целта, допирайки се към върховете на препятствията. Създава се неориентиран граф, където ръб е линеен сегмент, свързващ двойка взаимно видими възли. Два възела са взаимно видими, ако ръбът (отсечка), свързващ двата възела, не пресича нито един ръб от препятствие. В литературата се описват две разновидности на алгоритъма, като и в двата случая е необходимо първо да се определи свободното конфигурационното пространство. Метода най-често се прилага за мобилни роботи, а границите на конфигурационното пространство и препятствията е необходимо да могат да се представят с помощта на многоъгълници.

Предимства на планирането с граф на видимост.

- Приложим е за мобилни роботи в ясно определена среда с многоъгълни препятствия.
- Алгоритъмът се реализира лесно.

- Разбираемо и ясно се представя графично в двумерното пространство.
- Гарантирано дава най-краткия път в 2D.

Недостатъци на метода:

- Изисква препятствията да са многоъгълни и робота да е кръгъл или със значително по-малки размери от препятствията.
- Пътят е опасно близо до препятствията – няма място за грешни движения от робота.
- Не е приложим за конфигурационни пространства с повече от две измерения. Не се мащабира добре до 3D. Най-краткия път в 3D не е през върховете (виж фиг.11).
- Изисква реализиране на остри завои на едно място. Това го прави приложим само за част от мобилните роботи.
- Не е подходящ за стационарни роботи.

Клетъчна декомпозиция. Триъгълна и трапецовидна декомпозиция.

Съществуват методи за планиране на път, базирани на идеята за разделяне (декомпозиция) на конфигурационното пространство на множество клетки. Първо конфигурационното пространство се разделя на C_{free} и C_{obst} , като отново препятствията се представят с многоъгълници. Сега обаче свободното пространство се разделя на множество клетки, които покриват цялото C_{free} и се търси стратегия за придвижване между свободните клетки. Декомпозициите са два вида: точна декомпозиция – разделя свободното пространство на изпъкнали точни полигони и приблизителна декомпозиция.

Предимства на метода

- Гарантирано намира свободен път, ако съществува.
- Клетките се определят лесно в двумерното пространство.

Недостатъци

- Необходимо е да се изберат подходящи точки от клетките, през които да се придвижва робота без сблъсък.
- Не намира най-късия път.
- При повече от две измерения се прилага трудно.

Планиране на пътя с диаграма на Вороной (Voronoi Diagram)

Може да се интересуваме от това да намерим път от една точка в конфигурационното пространство до друга, без да искаме той да е най-краткия, а да е най-надеждния. За да сме сигурни, че няма да има конфликти с препятствия си поставяме за цел да се движим максимално далече от тях. На такова условие отговарят ръбовете от диаграма на Вороной (https://en.wikipedia.org/wiki/Voronoi_diagram). Диаграмата на Вороной е кръстена на математика Григорий Вороной (https://en.wikipedia.org/wiki/Georgy_Voronoy). През 1908г. Вороной формализира n-мерния случай за концепцията, с която сега познаваме диаграмите на Вороной. Накратко, диаграмата на Вороной записва информация за разстоянията между групи точки във многомерно пространство.

Предимства на метода с диаграма на Вороной:

- Гарантирано намира свободен път, ако съществува.
- Намереният път е надежден, защото траекторията е максимално отдалечена от препятствията.
- Приложим е и за конфигурационни пространства с повече от две измерения.

Недостатъци

- Необходимо е да се ползват специализирани алгоритми за генериране на границите на диаграмата на Вороной.
- Не намира най-късия път.
- При повече от две измерения се прилага трудно.

Потенциални полета.

Идеята за изкуствени потенциални полета за планиране на движението на робот е въведена за първи път от О. Khatib - 1985г. и е допълнително експлоатирана от редица учени (<https://profiles.stanford.edu/oussama-khatib>). В литературата се записва като **Artificial Potential Field (APF)**. Изкуственото потенциално поле е скаларна функция, дефинирана във всяка точка на ставното (конфигурационно) пространство на робота. При това два основни въпроса са важни за създаването на такава функция. Първо трябва да избягваме сблъсъци с препятствията и второ да се достига до целевата конфигурация от всяка зададена начална конфигурация, ако е възможно. Това е един от малкото методи за планиране на движения на робот в реално време. Обикновено градиентът на полето може да се изчисли бързо, така че движението може да се изчисли в реално време (реактивно управление) вместо предварително планиране. С подходящи сензори, методът може дори да се справи с препятствия, които се движат или се появяват неочаквано. Сред подходите за планиране на път изкуственото потенциално поле е много популярно поради многото си предимства като обща техника за навигация на роботи.

Някои от основните предимства са изброени по-долу.

- Прилага се както за стационарни така и за мобилни роботи;
- Метода е приложим за редундантни роботи;
- Може да се прилага в реално време;
- При ниска размерност се изпълнява и визуализира лесно, следователно резултантното поведение на робота може да се предскаже от дизайнера.
- Включва и отчита динамиката на робота.
- Поддържа паралелизъм; всяко от моделираните полета на препятствия и целта е независимо от останалите и може да бъде внедрени самостоятелно.
- Може лесно да се параметризира и конфигурира по време на фазата на проектиране или в реално време.
- Процесът на комбиниране е гъвкав и може да бъде променен, за да отрази различните приоритети.

Някой от по важните проблеми на метода могат да се обобщят:

- Потенциалното поле не трябва да показва повече минимума от минимума при целевата позиция, определена от конкретната задача на робота. Роботът може да се залепи в някой от местните минимума на потенциалното поле, далеч от целта, дори когато съществува осъществимо движение към целта. В някои случаи е възможно да се създаде потенциал за гарантиране, че единственият местен минимум е към целта, за да се премахне този проблем. Решаването на проблема за намиране на изкуствено потенциално поле без локални минимума често включва използването на така наречените хармонични функции. Хармоничните функции наистина нямат локални минимума, но са доста трудни за определяне при произволни гранични условия.

- Потенциалното поле около препятствията трябва вярно да затваря формата на препятствията дори за сложни неструктурирани повърхности на препятствия.
- За предпочитане е потенциалното поле да бъде конструирано в конфигурационното пространство на роботът, където роботът е представен от една (контролируема) точка, въпреки че препятствията съществуват във физическото пространство на робота. Трудно се конструира потенциалната енергия в конфигурационното пространство. Това е защото не е лесно да се представят препятствията в конфигурационното пространство. Звената на робота се преместват и завъртат при което постоянно трябва да се проверява за конфликти.
- Трудно се изчислява градиента в конфигурационното пространство. При многомерни пространства дискретното изчисляване на градиента изисква многократни изчисления за големината и посоката на векторите в отделните направления. Изчисляването на минималното разстояние до препятствие в конфигурационното пространство също е трудно.

6. Заключение

Това приложение обобщава работата на колектива на проекта свързана с дейност 2.2. Представен е анализ на методи за генериране на конфигурационно пространство на работи. Направена е класификация на алгоритми и подходи свързани с методи за планиране на движения на работи в среда с препятствия, които ползват конфигурационното пространство. Областта е голяма и се развива динамично и в този смисъл тук са представени основните базови концепции. По-пълна информация е представена в разработените лекции по „Планиране на движения в сложна среда“, предназначени за магистри от факултета по математика и информатика на СУ. Проучването и анализите са в основата за развитие на публикуваните до момента статии по проекта и ще се ползват за развитието му в етап 2.

Литература

1. Denavit, J.; Hartenberg, R. A kinematic notation for lower-pair mechanisms based on matrices. *ASME J. Appl. Mech.* 1951, 22, 215–221.
2. Kucuk, S.; Bingul, Z. Robot Kinematics: Forward and Inverse Kinematics. In *Industrial-Robotics-Theory-Modelling-Control*; Cubero, S., Ed.; Intech Open: Berlin, Germany, 2006.
3. Venkata Neeraj Kumar, R.; Sreenivasulu, R. Inverse Kinematics (IK) Solution of a Robotic Manipulator using PYTHON. *J. Mechatron. Robot.* 2019, 3, 542–551.
4. Ayush, G. A Geometric Approach to Inverse Kinematics of a 3 DOF Robotic Arm. *Int. J. Res. Appl. Sci. Eng. Technol.* 2018, 6, 3524–3530.]
5. Tokarz, K.; Kieltyka, S. Geometric approach to inverse kinematics for arm manipulator. In *Proceedings of the 14th WSEAS International Conference on Systems: Part of the 14th WSEAS CSCC Multi Conference—Volume II*, Corfu, Greece, 22–24 July 2010.
6. Wang, P.; Zhou, Y.; Yan, N. An Inverse Solution Algorithm for Industrial Robot. *J. Phys. Conf. Ser.* 2022, 2173, 012085.
7. Manseur, R.; Doty, K.L. A Fast Algorithm for Inverse Kinematic Analysis of Robot Manipulators. *Int. J. Robot. Res.* 1988, 7, 52–63.

8. Gan, J.Q.; Oyama, E.; Rosales, E.M.; Hu, H. A complete analytical solution to the inverse kinematics of the Pioneer 2 robotic arm. *Robotica* 2005, 23, 123–129.
9. Li, Y.; Wang, L. Kinematic Model and Redundant Space Analysis of 4-DOF Redundant Robot. *Mathematics* 2022, 10, 574.
10. Ghosal, A. Resolution of redundancy in robots and in a human arm. *Mech. Mach. Theory* 2018, 125, 126–136.
11. Xie, S.; Sun, L.; Wang, Z.; Chen, G. A speedup method for solving the inverse kinematics problem of robotic manipulators. *Int. J. Adv. Robot. Syst.* 2022.
12. Heidari, O.; Gracia, A.P. Inverse Kinematics Using a Converging Paths Algorithm. *Adv. Robot Kinemat.* 2020, 2020, 15.
13. Alavandar, S.; Nigam, M.J. Inverse Kinematics Solution of 3DOF Planar Robot using ANFIS. *Int. J. Comput. Commun. Control* 2008, 3, 150–155.
14. Jamali, A.; Khan, R.; Rahman, M.M. A new geometrical approach to solve inverse kinematics of hyper redundant robots with variable link length. In *Proceedings of the 4th International Conference on Mechatronics (ICOM)*, Kuala Lumpur, Malaysia, 17–19 May 2011.
15. Al-Mashhadany, Y.I. Inverse Kinematics Problem (IKP) of 6-DOF Manipulator by Locally Recurrent Neural Networks (LRNNs). In *Proceedings of the International Conference on Management and Service Science*, Wuhan, China, 24–26 August 2010.
16. Semwal, V.B.; Gupta, Y. Performance Analysis of Data-Driven Techniques for Solving Inverse Kinematics Problems. *Intell. Syst. Appl.* 2021, 294, 85–99.
17. Köker, R.; Çakar, T.; Sari, Y. A neural-network committee machine approach to the inverse kinematics problem solution of robotic manipulators. *Eng. Comput.* 2014, 30, 641–649.
18. Kenwright, B. Inverse Kinematics—Cyclic Coordinate Descent (CCD). *J. Graph. Tools* 2012, 16, 177–217.
19. Chen, Y.; Luo, X.; Han, B.; Jia, Y.; Liang, G.; Wang, X. A General Approach Based on Newton's Method and Cyclic Coordinate Descent Method for Solving the Inverse Kinematics. *Appl. Sci.* 2019, 9, 5461.
20. Song, W.; Hu, G. A Fast Inverse Kinematics Algorithm for Joint Animation. *Procedia Eng.* 2011, 24, 350–354.
21. Müller-Cajar, R.; Mukundan, R. Triangulation: A New Algorithm for Inverse Kinematics. *Proc. Image Vis. Comput.* 2007, 181–186. Available online: https://ir.canterbury.ac.nz/bitstream/handle/10092/743/12607089_ivcnz07.pdf;sequence=1 (accessed on 18 September 2022).
22. Panchanand, J.; Biswal, B.B. Inverse Kinematic Solution of 5R Manipulator Using ANN and ANFIS. *Int. J. Robot. Autom. IJRA* 2015, 4, 109–123.
23. Takehiko, O.; Hajime, K. Solution for Ill-Posed Inverse Kinematics of Robot Arm by Network Inversion. *J. Robot.* 2010, 2010.
24. Steven M. LaValle, *PLANNING ALGORITHMS*, Cambridge University Press, 2006.
25. Sertac Karaman and Emilio Frazzoli. *Sampling-based Algorithms for Optimal Motion Planning*. The International Journal of Robotics Research, 2011.
26. K. Hughes, A. Tokuta, and R. Ranganathan. Trulla: An algorithm for path planning among weighted regions by localized propagations. In *Proceedings of IEEE Conference on Intelligent Robots and Systems*, 1992.
27. Dunias, P. (1996). Autonomous robots using artificial potential fields. Technische Universiteit Eindhoven. <https://doi.org/10.6100/IR470384>
28. Artificial Potential Field and Feature Extraction Method for Mobile Robot Path Planning in Structured Environments.

29. Kevin M. Lynch and Frank C. Park, Modern Robotics Mechanics, Planning, and Control
30. Leng-feng Lee, Decentralized Motion Planning Within an Artificial Potential Framework (APF) for Cooperative Payload Transport by Multi-robot Collectives, master of science, 2004, Department of Mechanical and Aerospace Engineering, State University of New York at Buffalo, Buffalo, New York 14260.
31. O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," *International Journal of Robotic Research*, vol. 5, no. 1, pp. 90, 1986.
32. R. Volpe and P. Khosla, "Manipulator control with superquadric artificial potential functions: theory and experiments," in *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 20, no. 6, pp. 1423-1436, Nov.-Dec. 1990, doi: 10.1109/21.61211.
33. S. S. GE and Y. J. CUI, "Dynamic motion planning for mobile robots using potential field method," *Autonomous Robots*, vol. 13, pp. 207-222, 2002.