

Programowanie animacji

Krzysztof Gdawiec



UNIWERSYTET ŚLĄSKI
INSTYTUT INFORMATYKI

Łańcuchy kinematyczne

Często przydaje się opisywanie ruchu obiektu względem innego obiektu, np. ruch Księżyca wokół Słońca. O wiele łatwiej jest opisać ruch Księżyca wokół Ziemi, a Ziemi wokół Słońca niż opisywać bezpośrednio ruch Księżyca wokół Słońca.

Inne przykłady możemy znaleźć w robotyce, silnikach spalinowych, animacji postaci ludzkich.

Układ obiektów poruszających się względem innych obiektów, których ruch możemy łatwo opisać nazywać będziemy **hierarchią ruchu**.

Jeśli elementy są fizycznie połączone, to takie połączenie nazywamy **parą kinematyczną**, a układy połączonych obiektów **łańcuchem kinematycznym** lub krócej łańcuchem.

Model ze **zredukowanym wymiarem** jest to model, w którym struktura hierarchii ruchu wymusza nałożenie ograniczeń i wymaga podania mniejszej liczby parametrów niż liczba parametrów potrzebnych do określenia położenia w inny sposób.

Pozycja Księżyca względem Ziemi może być określona przez jeden parametr (kąt) ponieważ Księżyc obraca się wokół Ziemi w ustalonej płaszczyźnie i w stałej odległości. Płaszczyzna obrotu i odległość mogą być wbudowane w model hierarchiczny.

Pozycja Księżyca względem Ziemi może być określona przez jeden parametr (kąt) ponieważ Księżyc obraca się wokół Ziemi w ustalonej płaszczyźnie i w stałej odległości. Płaszczyzna obrotu i odległość mogą być wbudowane w model hierarchiczny.

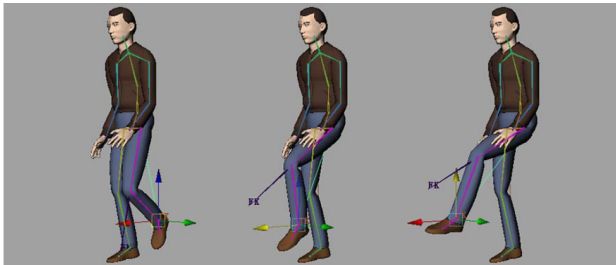
Łańcuchy kinematyczne są powszechnie stosowane do animowania postaci ludzkich. Ruch ich kończyn jest określony za pomocą hierarchii ruchów obrotowych sztywnych obiektów połączonych stawami.

Mamy dwa podejścia do ustalania położeń obiektów tworzących hierarchię:

- ▶ kinematyka prosta (ang. forward kinematics) – podajemy wartości parametrów (kąty rozwarcia stawów),



- ▶ kinematyka odwrotna (ang. inverse kinematics) – podajemy pożądaną połozenie np. dłoni, a system znajduje odpowiadające mu wartości parametrów



Modelowanie hierarchiczne

Modelowanie hierarchiczne polega na wymuszeniu ograniczeń nałożonych na względne położenia obiektów zorganizowanych w drzewopodobną strukturę danych.

Modelowanie hierarchiczne

Modelowanie hierarchiczne polega na wymuszeniu ograniczeń nałożonych na względne położenia obiektów zorganizowanych w drzewopodobną strukturę danych.

W grafice powszechnie występującym modelem hierarchicznym jest model zawierający obiekty połączone końcami. Takie modele są przydatne w modelowaniu ludzi i zwierząt.

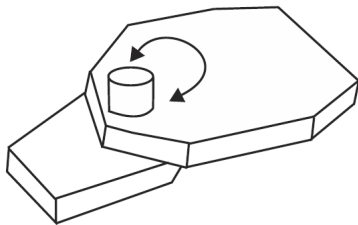
Postać, dla której manipulujemy stawami kończyn w celu otrzymania ruchu nazywamy **postacią artykułowaną**, a ruch kończyny osiągnięty przez zmianę kąta rozwarcia stawu **artykulacją**.

W robotyce, która była podstawą animowania hierarchii układy obiektów połączonych w pary kinematyczne nazywane są **manipulatorami**. Obiekty wchodzące w ich skład nazywane są **członami**, a wolny koniec łańcucha złożonego z członów i przegubów jest nazywany **efektorem końcowym**.

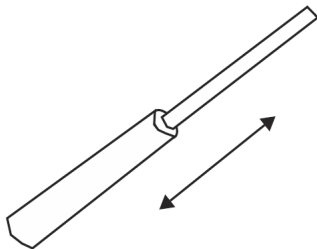
W robotyce, która była podstawą animowania hierarchii układy obiektów połączonych w pary kinematyczne nazywane są **manipulatorami**. Obiekty wchodzące w ich skład nazywane są **członami**, a wolny koniec łańcucha złożonego z członów i przegubów jest nazywany **efektorem końcowym**.

Przegubem nazywamy parę kinematyczną, w której jeden człon obraca się względem drugiego wokół ustalonego punktu.

Para przesuwna jest to para, która umożliwia przesuwanie jednego członu względem drugiego.



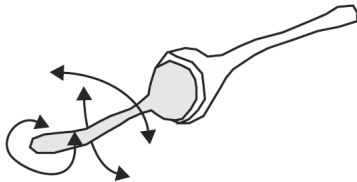
Revolute joint



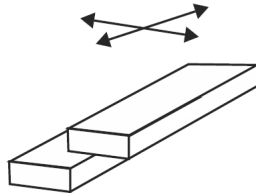
Prismatic joint

Tego typu pary kinematyczne umożliwiają ruch opisany jednym parametrem liczbowym co określamy mianem **jednego stopnia swobody**.

Pary kinematyczne mające więcej niż jeden stopień swobody nazywamy **parami złożonymi**. Przykładami są: przegub kulkowy, para płaska.

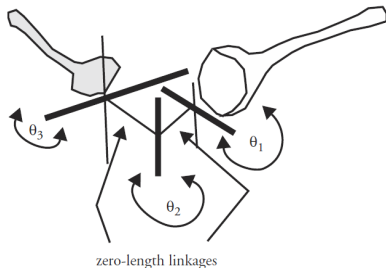


Ball-and-socket joint

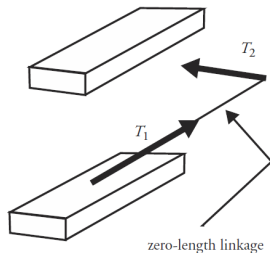


Planar joint

W pewnych przypadkach pary kinematyczne o $n > 1$ stopniach swobody (np. przeguby kulkowe) są modelowane za pomocą układu n par o jednym stopniu swobody i $n - 1$ członach o długości 0.



Ball-and-socket joint modeled as 3 one-degree joints with zero-length links



Planar joint modeled as 2 one-degree prismatic joints with zero-length links

W innym rozwiązaniu pary te opisywane są za pomocą parametrów wektorowych, takich jak układy kątów Eulera lub kwaterniony.

Struktury danych dla modelowania hierarchicznego

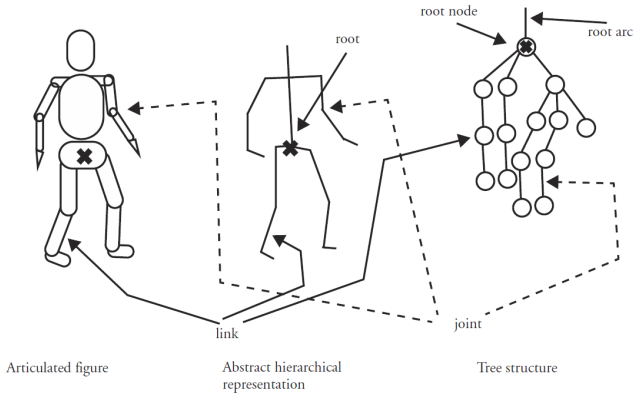
Postacie ludzkie i zwierząt wygodnie modelować jest za pomocą łańcuchów hierarchicznych. Łańcuchy takie reprezentowane są przez strukturę **wierzchołków** (ang. nodes) połączonych **krawędziami** (ang. arcs).

Najwyższy wierzchołek jest **korzeniem** drzewa i reprezentuje główny człon łańcucha. Pozycja tego członu jest określona w globalnym układzie współrzędnych.

Położenia członów reprezentowanych przez pozostałe wierzchołki są określone względem korzenia.

Wierzchołek, z którego nie wychodzą żadne krawędzie nazywamy **liściem**.

Odwzorowanie hierarchii na drzewo wiąże każdy wierzchołek z członem łańcucha kinematycznego, a krawędź reprezentującą parę kinematyczną z informacją o przekształceniu, któremu będą poddane wszystkie człony łańcucha położone niżej w hierarchii.



W strukturze drzewa wprowadzamy krawędź do korzenia reprezentującą globalne przekształcenie, któremu należy poddać główny człon.

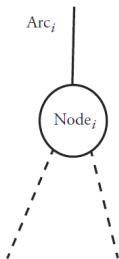
Zmianie tego przekształcenia w sposób sztywny odpowiada przemieszczenie całej struktury w globalnym układzie współrzędnych.

Wierzchołek i zawiera:

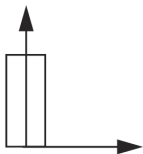
- ▶ przekształcenie umieszczające obiekt tak, aby jego środek obrotu znalazł się w początku układu (opcjonalne),
- ▶ dane opisujące obiekt.

Krawędź i zawiera:

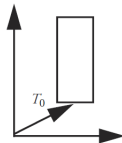
- ▶ stałe przekształcenie umieszczające człon i w pozycji wyjściowej względem członu $i - 1$,
- ▶ zmienne przekształcenie odpowiadające za artykulację członu i .



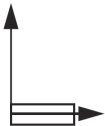
Rozważmy przykład dwuwymiarowego łańcucha kinematycznego złożonego z trzech członów.



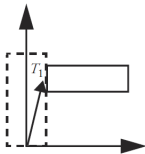
Original definition of root object (Link 0)



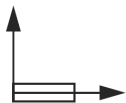
Root object (Link 0) transformed (translated and scaled) by T_0 to some known location in global space



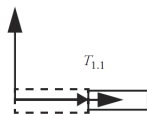
Original definition of Link 1



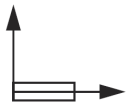
Link 1 transformed by T_1 to its position relative to untransformed Link 0



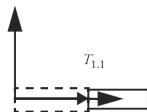
Original definition of Link 1.1



Link 1.1 transformed by $T_{1,1}$
to its position relative to
untransformed Link 1

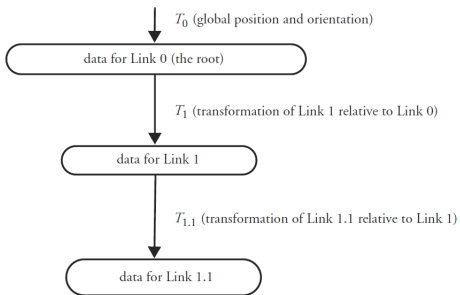


Original definition of Link 1.1



Link 1.1 transformed by $T_{1,1}$
to its position relative to
untransformed Link 1

Zależności możemy reprezentować za pomocą drzewa.



Krawędź drzewa zawiera informacje o przekształceniu, któremu należy poddać obiekt reprezentowany przez wierzchołek, do którego ta krawędź wchodzi.

Przekształceniu temu poddawane są również pozostałe człony położone niżej w hierarchii.

Krawędź drzewa zawiera informacje o przekształceniu, któremu należy poddać obiekt reprezentowany przez wierzchołek, do którego ta krawędź wchodzi.

Przekształceniu temu poddawane są również pozostałe człony położone niżej w hierarchii.

Jeśli V_0 jest wierzchołkiem bryły będącej członem C_0 (danym w lokalnym układzie tej bryły), to współrzędne w globalnym układzie V'_0 wyrażają się wzorem:

$$V'_0 = T_0 V_0.$$

Współrzędne wierzchołka V_1 bryły członu C_1 w globalnym układzie wyrażają się wzorem:

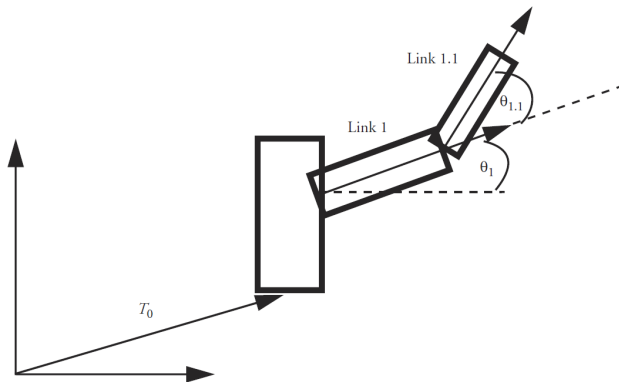
$$V'_1 = T_0 T_1 V_1.$$

Podobnie współrzędne wierzchołka $V_{1.1}$ bryły członu $C_{1.1}$ w globalnym układzie wyrażają się wzorem:

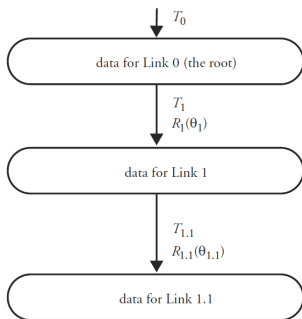
$$V'_{1.1} = T_0 T_1 T_{1.1} V_{1.1}.$$

Zatem podczas przechodzenia przez wierzchołki drzewa nowe przekształcenie związane z krawędzią jest składane z przekształceniem, któremu poddawany jest wierzchołek, z którego ta krawędź wychodzi.

Dodajmy teraz możliwość obracania przegubami.



Obrót przegubu w drzewie związany jest z krawędzią wchodzącą do wierzchołka reprezentującego człon, który ma być obracany.



Obrót należy wykonać przed zastosowaniem stałego przekształcenia związanego z krawędzią. Jeśli dla wierzchołka ponadto mamy przekształcenie, to należy je wykonać najpierw, następnie obrót, a na końcu przekształcenie stałe krawędzi.

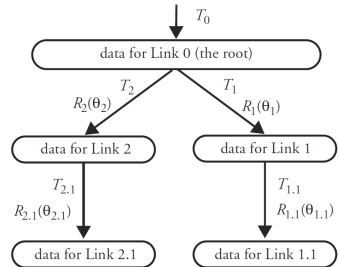
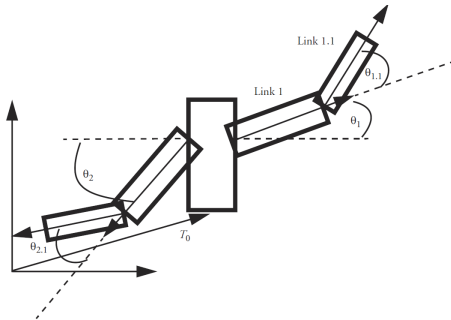
Współrzędne wierzchołka V_1 członu C_1 w globalnym układzie wyrażają się wzorem:

$$V'_1 = T_0 T_1 R_1(\theta_1) V_1,$$

zaś współrzędne wierzchołka $V_{1.1}$ członu $C_{1.1}$:

$$V'_{1.1} = T_0 T_1 R_1(\theta_1) T_{1.1} R_{1.1}(\theta_{1.1}) V_{1.1}$$

Jeśli istnieje druga kończyna to struktura drzewa musi odzwierciedlić jej obecność.



Kinematyka prosta

Animacja łańcucha kinematycznego odbywa się przez wykonanie działań na parametrach przekształceń par kinematycznych (w przykładzie były to obroty). Na podstawie tych parametrów konstruowane są macierze przekształceń związanych z krawędziami drzewa.

Pełny zestaw parametrów określający położenia wszystkich części obiektu nazywany jest **pozycją** (ang. pose). Z parametrów tych tworzony jest wektor (**wektor pozycji**); każda jego współrzędna odpowiada jednemu stopniowi swobody.

W prostej animacji użytkownik określa pozycje dla klatek kluczowych w sposób interaktywny, a następnie dokonywana jest interpolacja parametrów dla klatek pośrednich.

Ustawienie obiektu w pożądanym położeniu przez dobieranie wartości parametrów może być żmudne oraz często wykonywane jest metodą prób i błędów.

Określanie położenia części obiektu przez zadawanie wartości parametrów nazywane jest **kinematyką prostą**.

W algorytmie kinematyki prostej przeszukujemy drzewo hierarchii w głąb zaczynając od korzenia. Przy przetwarzaniu krawędzi należy zapamiętywać i przywracać przekształcenia do tego celu w algorytmie stosuje się stos macierzy przekształceń.

Znaczenie poszczególnych zmiennych dla krawędzi:

- ▶ `nodePtr` – wskaźnik wierzchołka reprezentującego dane artykułowane przez krawędź,
- ▶ `Lmatrix` – macierz przekształcenia umieszczającego człon reprezentowany przez syna w pozycji wyjściowej względem ojca,
- ▶ `Amatrix` – macierz przekształcenia realizującego artykulację syna; animacja dokonywana jest przez zmienianie tej macierzy,
- ▶ `arcPtr` – wskaźnik do kolejnej krawędzi wychodzącej z tego samego wierzchołka; dla ostatniej krawędzi ma wartość `NULL`.

Znaczenie poszczególnych zmiennych dla wierzchołka:

- ▶ `dataPtr` – dane opisujące model geometryczny odpowiedniego członu,
- ▶ `Tmatrix` – macierz przekształcenia umieszczającego człon w pozycji wyjściowej do artykulacji (np. umieszcza środek obrotu w początku układu współrzędnych),
- ▶ `arcPtr` – wskaźnik do pierwszej krawędzi wychodzącej z wierzchołka.

Przetwarzanie hierarchii zaczynamy od krawędzi wchodzącej do korzenia oraz macierzy jednostkowej:

```
traverse( rootArcPtr, I )
```

Algorytm 1: Algorytm kinematyki prostej.

```
1 traverse( arcPtr, matrix )
  /* weź przekształcenia związane z krawędzią i złóż je z macierzą
   * bieżącego przekształcenia */
2   matrix = matrix * arcPtr → Lmatrix
3   matrix = matrix * arcPtr → Amatrix
  // przetwórz dane związane z wierzchołkiem
4   nodePtr = arcPtr → nodePtr
5   Push( matrix )
6   matrix = matrix * nodePtr → Tmatrix
7   articulatedData = transformData( matrix, nodePtr → dataPtr )
8   draw( articulatedData )
9   Pop( matrix )
  // przetwórz synów wierzchołka
10  if nodePtr → arcPtr != NULL then
11    nextArcPtr = nodePtr → arcPtr
12    while nextArcPtr != NULL do
13      Push( matrix )
14      traverse( nextArcPtr, matrix )
15      Pop( matrix )
16      nextArcPtr = nextArcPtr → arcPtr
```

Kinematyka odwrotna

W kinematyce odwrotnej użytkownik podaje pożądane położenie i ewentualne zorientowanie efektora końcowego oraz początkowy wektor pozycji.

Na podstawie tych danych obliczane są wartości parametrów (końcowy wektor pozycji) zapewniające osiągnięcie podanego położenia.

Problem znalezienia tych wartości może mieć zero, jedno lub wiele rozwiązań.

Jeśli ograniczenia, tzw. **więzy**, nałożone na łańcuch powodują brak rozwiązań, to łańcuch nazywamy **przesztywnionym**.

Jeśli więzów jest za mało, przez co postawiony problem ma wiele rozwiązań, to łańcuch jest **niedosztywniony**.

Przestrzeń osiągalna jest to obszar, w którym efektor końcowy może się znaleźć.

Przestrzeń o pełnym dostępie to obszar, w którym efektor końcowy może się znaleźć bez ograniczania jego położenia kąтового.

Po obliczeniu wartości parametrów par kinematycznych obiekt może być animowany przez interpolację wektorów pozycji początkowej i końcowej.

Jeśli wektory te różnią się znacznie, to nie ma możliwości dokładnej kontroli nad ścieżką, po której porusza się obiekt.

W takim przypadku można określić ciąg położeń pośrednich efektora końcowego, a następnie użyć ich jako danych w zadaniach kinematyki odwrotnej. W ten sposób efektor końcowy będzie się poruszał po ścieżce określonej przez animatora.

Jeśli łańcuch jest prosty, to parametry par kinematycznych mogą być obliczone analitycznie.

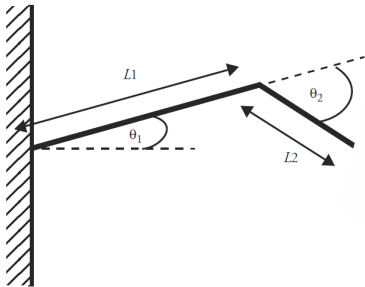
Jeśli łańcuch jest prosty, to parametry par kinematycznych mogą być obliczone analitycznie.

Jeśli łańcuch jest skomplikowany, to można zastosować metodę przyrostową. W metodzie tej wyznaczana jest macierz (tzw. jacobian) opisująca wpływ zmian wartości par kinematycznych na zmiany położenia i zorientowania efektora końcowego.

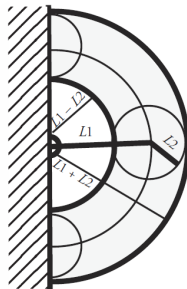
Efektor końcowy jest iteracyjnie popychany do chwili, gdy znajdzie się w granicach dopuszczalnego błędu od położenia końcowego.

Rozwiązanie analityczne prostego zadania

Rozważmy proste dwuczłonowe ramię w przestrzeni 2D, z dwoma obrotowymi stopniami swobody. Oznaczmy długości członów przez L_1 i L_2 .



Configuration



Reachable workspace

Będziemy korzystać z twierdzenia
kosinusów.

$$\cos \theta_T = \frac{X}{\sqrt{X^2 + Y^2}}$$

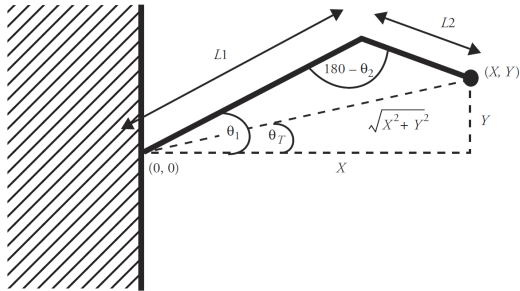
$$\theta_T = \arccos \left(\frac{X}{\sqrt{X^2 + Y^2}} \right)$$

$$\cos(\theta_1 - \theta_T) = \frac{L_1^2 + X^2 + Y^2 - L_2^2}{2L_1\sqrt{X^2 + Y^2}}$$

$$\theta_1 = \arccos \left(\frac{L_1^2 + X^2 + Y^2 - L_2^2}{2L_1\sqrt{X^2 + Y^2}} \right) + \theta_T$$

$$\begin{aligned} \cos(180 - \theta_2) &= -\cos \theta_2 = \\ &= \frac{L_1^2 + L_2^2 - (X^2 + Y^2)}{2L_1L_2} \end{aligned}$$

$$\theta_2 = \arccos \left(\frac{X^2 + Y^2 - L_1^2 - L_2^2}{2L_1L_2} \right)$$



Zanim jednak zrobimy obliczenia z poprzedniego slajdu musimy sprawdzić czy:

$$|L_1 - L_2| \leq \sqrt{X^2 + Y^2} \leq L_1 + L_2.$$

Rozważane zadanie ma dwa rozwiązania. Odpowiadają im położenia członów ramienia symetryczne względem prostej przechodzącej przez $(0, 0)$ i (X, Y) .

W bardziej skomplikowanych przypadkach może istnieć nieskończenie wiele rozwiązań.

Jakobian

Założmy, że mamy $f_i : \mathbb{R}^n \rightarrow \mathbb{R}$ dla $i = 1, \dots, m$. Zatem

$$y_i = f_i(x_1, x_2, \dots, x_n)$$

dla $i = 1, \dots, m$ oraz $(x_1, x_2, \dots, x_n)^T \in \mathbb{R}^n$.

Do opisanie szybkości zmian zmiennych wyjściowych wywołanych zmienianiem wartości zmiennych x_i możemy użyć różniczek.

Z zasady łańcuchowej różniczkę y_i możemy wyrazić za pomocą różniczek zmiennych x_j :

$$\partial y_i = \frac{\partial f_i}{\partial x_1} \partial x_1 + \frac{\partial f_i}{\partial x_2} \partial x_2 + \dots + \frac{\partial f_i}{\partial x_n} \partial x_n.$$

Wzory z poprzedniego slajdu możemy również zapisać w postaci macierzowej:

$$Y = F(X),$$
$$\partial Y = \frac{\partial F}{\partial X} \partial X.$$

Macierz $J = \frac{\partial F}{\partial X}$ o wymiarach $m \times n$ nazywamy **jakobianem**. Jest ona funkcją zmiennych x_i czyli $J = J(X)$.

O jakobianie można myśleć jak o odwzorowaniu szybkości zmian zmiennych x_i na szybkości zmian y_j :

$$\dot{Y} = J(X)\dot{X}.$$

Określając jacobian dla łańcucha kinematycznego za zmienne niezależne x_i przyjmujemy parametry par kinematycznych, a zmienne y_j będą opisywać położenie przestrzenne i kątowe efektora końcowego:

$$Y = [p_x, p_y, p_z, \alpha_x, \alpha_y, \alpha_z]^T$$

W tym przypadku jacobian wiąże wektor prędkości kątowych obrotu przegubów $\dot{\theta}$ z wektorem \dot{Y} reprezentującym prędkość ruchu postępowego i obrotowego efektora końcowego:

$$\dot{Y} = J(\theta)\dot{\theta}.$$

Oznaczmy $V = \dot{Y}$. Wektor ten reprezentuje pożądaną zmianę położenia efektora końcowego. Zmiana ta określona jest przez różnicę między pożądanym i bieżącym położeniem.

Prędkości w przestrzeni są wektorami o trzech współrzędnych:

$$V = [v_x, v_y, v_z, \omega_x, \omega_y, \omega_z]^T.$$

Wektor $\dot{\theta}$ składa się z prędkości kątowych ruchu obrotowego przegubów, które są niewiadomymi:

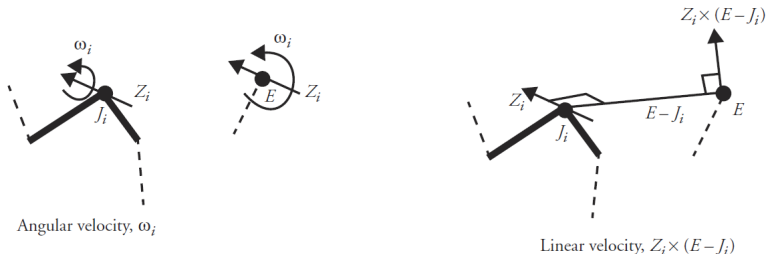
$$\dot{\theta} = [\dot{\theta}_1, \dots, \dot{\theta}_n]^T.$$

Macierz J ma postać:

$$J = \begin{bmatrix} \frac{\partial p_x}{\partial \theta_1} & \frac{\partial p_x}{\partial \theta_2} & \cdots & \frac{\partial p_x}{\partial \theta_n} \\ \frac{\partial p_y}{\partial \theta_1} & \frac{\partial p_y}{\partial \theta_2} & \cdots & \frac{\partial p_y}{\partial \theta_n} \\ \vdots & \vdots & \cdots & \vdots \\ \frac{\partial \alpha_z}{\partial \theta_1} & \frac{\partial \alpha_z}{\partial \theta_2} & \cdots & \frac{\partial \alpha_z}{\partial \theta_n} \end{bmatrix}$$

Podczas obracania pojedynczego przegubu oś obrotu i prędkość kątowna tego obrotu opisują ruch efektora końcowego. Zmieniając parametr pary przesuwnej nie zmieniamy położenia kątownego efektora końcowego.

Kierunek ruchu postępowego efektora końcowego otrzymujemy obliczając iloczyn wektorowy wektora osi obrotu i wektora łączącego środek obrotu z położeniem efektora.

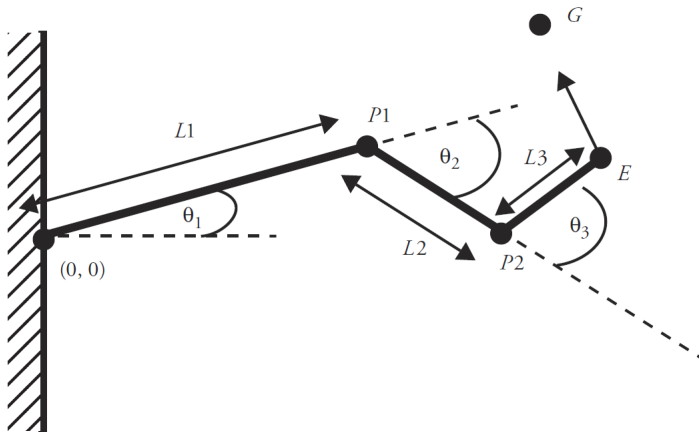


Pożądaną prędkość efektora końcowego są obliczane na podstawie różnicy parametrów opisujących jego pożądaną i bieżącą pozycję.

Obliczając jacobian należy opisać położenia członów łańcucha w tym samym układzie współrzędnych.

Często jednak informacja związana z parą kinematyczną jest podana w lokalnym układzie współrzędnych. Zatem należy dokonać przejścia do pewnego ustalonego układu współrzędnych np. globalnego lub związanego z efektem końcowym.

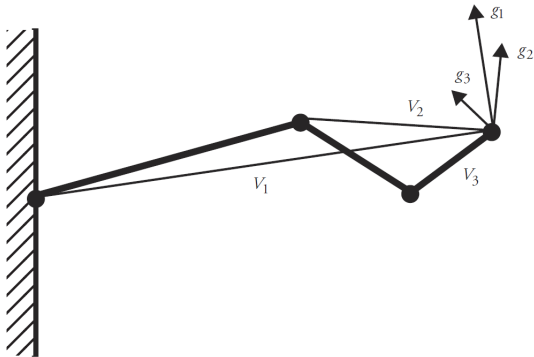
Rozważmy prosty przykład płaskiego manipulatora o trzech przegubach. Celem jest umieszczenie efektora końcowego w punkcie G .



Położenie kątowne efektora końcowego nie ma znaczenia.

Oś obrotu dla każdego przegubu jest prostopadła do płaszczyzny rysunku.

Prędkości efektora końcowego podczas obracania przegubów:



Pożądana prędkość ruchu efektora końcowego:

$$V = G - E.$$

Jakobian:

$$J = \begin{bmatrix} ([0, 0, 1] \times E)_x & ([0, 0, 1] \times (E - P_1))_x & ([0, 0, 1] \times (E - P_2))_x \\ ([0, 0, 1] \times E)_y & ([0, 0, 1] \times (E - P_1))_y & ([0, 0, 1] \times (E - P_2))_y \\ ([0, 0, 1] \times E)_z & ([0, 0, 1] \times (E - P_1))_z & ([0, 0, 1] \times (E - P_2))_z \end{bmatrix}$$

Rozwiązywanie numeryczne zagadnień kinematyki odwrotnej

Po obliczeniu jakobianu należy rozwiązać równanie:

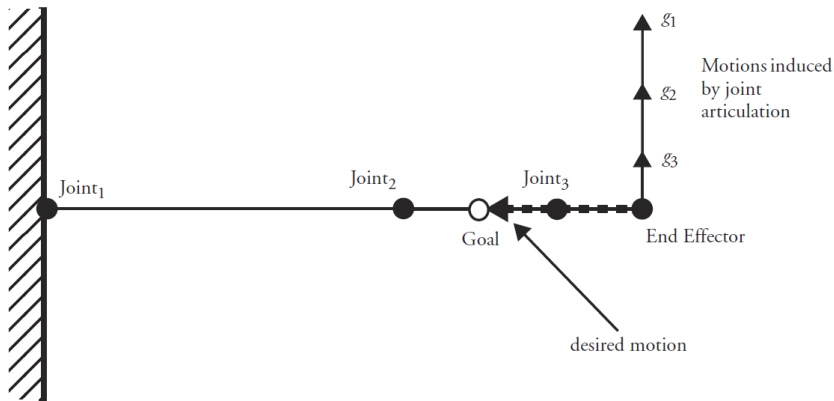
$$V = J\dot{\theta}.$$

Jeśli macierz J jest kwadratowa i nieosobliwa, to rozwiązanie można wyrazić wzorem:

$$\dot{\theta} = J^{-1}V.$$

Jeśli odwrotność jakobianu nie istnieje, to mówimy, że układ ma osobliwość dla odpowiednich kątów ustawienia przegubów.

Taka osobliwość może wykluczyć istnienie wektora prędkości obracania przegubów zapewniającej odpowiedni ruch efektora końcowego.



Zmiana kąta rozwarcia każdego przegubu daje wektor prędkości efektora końcowego prostopadły do pożądanego kierunku ruchu.

Żadna kombinacja liniowa takich wektorów nie daje ruchu efektora we właściwą stronę.

Zmiana kąta rozwarcia każdego przegubu daje wektor prędkości efektora końcowego prostopadły do pożądanego kierunku ruchu.

Żadna kombinacja liniowa takich wektorów nie daje ruchu efektora we właściwą stronę.

Również położenie łańcucha w pobliżu osobliwości może powodować kłopoty. Po niewielkim zaburzeniu kątów rozwarcia przegubów mamy układ nieosobliwy.

Kombinacja liniowa wektorów prędkości efektora końcowego ma bardzo duże współczynniki. Oznacza to ogromne prędkości obracania przegubów w pobliżu osobliwości.

Problemy z osobliwościami mogą być mniejsze jeśli liczba stopni swobody manipulatora jest większa niż liczba warunków, które należy spełnić.

W takim przypadku jacobian nie jest macierzą kwadratową i zadanie kinematyki odwrotnej może mieć nieskończenie wiele rozwiązań.

Problemy z osobliwościami mogą być mniejsze jeśli liczba stopni swobody manipulatora jest większa niż liczba warunków, które należy spełnić.

W takim przypadku jacobian nie jest macierzą kwadratową i zadanie kinematyki odwrotnej może mieć nieskończenie wiele rozwiązań.

Ponieważ jacobian nie jest macierzą kwadratową nie istnieje jej odwrotność w zwykłym sensie.

Jeśli wiersze jacobianu są liniowo niezależne (mówimy, że J jest wierszowo regularna), to macierz kwadratowa JJ^T jest nieosobliwa i możemy użyć macierzy $J^+ = J^T(JJ^T)^{-1}$.

Macierz J^+ nazywamy **pseudoodwrotnością** macierzy J .

Wektor $\dot{\theta}$ obliczamy następująco:

$$\dot{\theta} = J^+ V = J^T (JJ^T)^{-1} V.$$

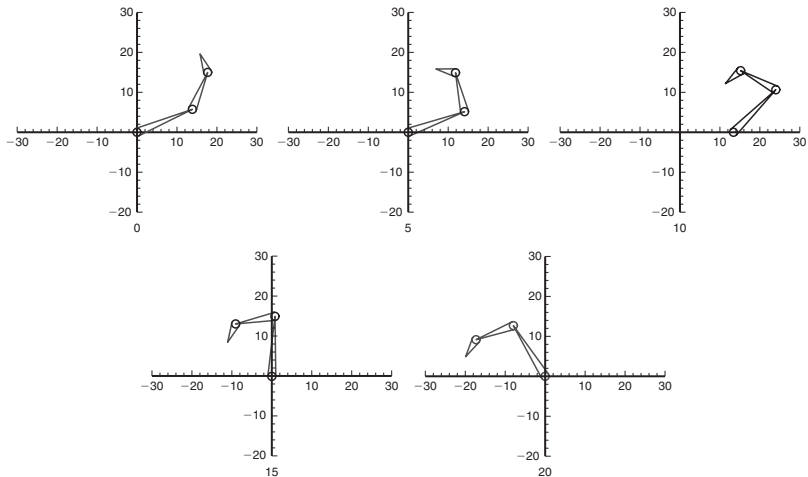
Wektor $\dot{\theta}$ obliczamy następująco:

$$\dot{\theta} = J^+ V = J^T (JJ^T)^{-1} V.$$

Zobaczmy przykład dwuwymiarowego łańcucha kinematycznego o trzech członach, które mają długości odpowiednio 15, 10, 5.

Początkową pozycję przyjmujemy jako $[\pi/8, \pi/4, \pi/4]$, a końcowe położenie w punkcie $(-20, 5)$.

Klatki 0, 5, 10, 15, 20.



Nawet gdy łańcuch jest niedosztywniony osobliwości mogą sprawiać kłopoty. W literaturze zaproponowano tzw. regularyzację.

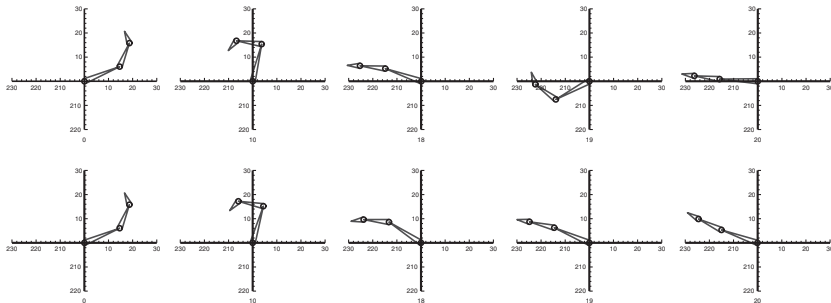
Wektor $\dot{\theta}$ w tej metodzie obliczany jest następująco:

$$\dot{\theta} = J^T (JJ^T + \lambda^2 I)^{-1} V,$$

gdzie λ jest to parametr określany przez użytkownika.

Zobaczmy ten sam przykład co poprzednio tylko tym razem położenie końcowe ustawiamy na $(-35, 5)$ czyli punkt poza zasięgiem efektora końcowego.

Klatki: 0, 10, 18, 19, 20. Pseudoodwrotność (góra), metoda z regularyzacją (dół).



Przy wyznaczaniu wektora $\dot{\theta}$ możemy również skorzystać jedynie z transponowanego jacobianu, tzn.

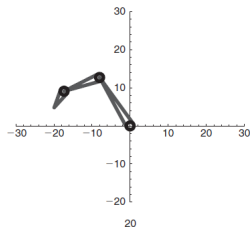
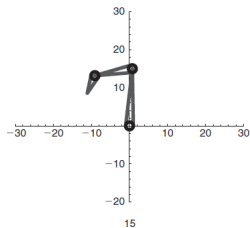
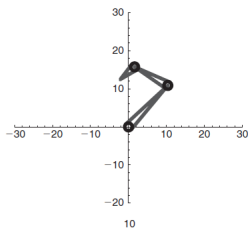
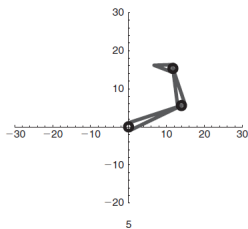
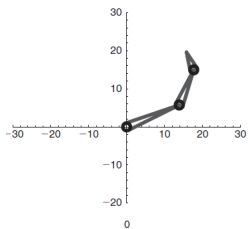
$$\dot{\theta} = \alpha J^T V,$$

gdzie α jest parametrem skalującym podawanym przez użytkownika.

Użycie transpozycji jacobianu pozwala uniknąć kosztów związanych z rozwiązywaniem układu równań liniowych, ale w pewnych przypadkach można otrzymać wektor zerowy.

Główną wadą jest fakt, że choć można w ten sposób przybliżyć efektor końcowy do celu, można go również od celu oddalić. Ponadto użytkownik musi podać dodatkowy parametr.

Położenie docelowe $(-20, 5)$. Klatki: 0, 5, 10, 15, 20.



Przy rozwiązywaniu problemu kinematyki odwrotnej możemy też skorzystać z np. metody cyklicznej modyfikacji współrzędnych.

W metodzie tej przetwarzamy jeden przegub na raz, kolejno od początku do końca łańcucha kinematycznego. Dla każdego przegubu wybierany jest kąt, który najbardziej przybliży efektor końcowy do celu.

Położenie docelowe $(-20, 5)$. Klatki: 0, 5, 10, 15, 20.

