

KAUNO TECHNOLOGIJOS UNIVERSITETAS  
INFORMATIKOS FAKULTETAS  
MULTIMEDIJOS INŽINERIJOS KATEDRA

Vilmantė Juteikienė

**Žmogaus judėjimo metodų  
kompiuterinėje animacijoje tyrimas**

Magistro darbas

Darbo vadovas

Doc. dr. A. Lenkevičius

Kaunas, 2006

KAUNO TECHNOLOGIJOS UNIVERSITETAS  
INFORMATIKOS FAKULTETAS  
MULTIMEDIJOS INŽINERIJOS KATEDRA

Vilmantė Juteikienė

**Žmogaus judėjimo metodų  
kompiuterinėje animacijoje tyrimas**

Magistro darbas

Kalbos konsultantė

Lietuvių k. katedros lekt.  
dr. I. Mickienė

2006 05

Vadovas

doc. dr. A. Lenkevičius

2006 05

Recenzentas

doc. prof. E. Bareiša

2006 05

Atliko

IFM-9/1 gr. stud.  
Vilmantė Juteikienė

2006 05

Kaunas, 2006

## **Research of human motion methods in computer animation**

### **SUMMARY**

Character's animation is the most complicated, but at the same time the most important field of the computer animation. The main problem is related with realistic, flexible representation of human motion, when complex and large computations occur.

Hybrid, human motion analyzing algorithm has been introduced theoretically. This algorithm is based on input of constraints, and one of the main constrains is mass distribution of human parts. Transitional motion representations are build, and their adjustment are computed while using aproximation of multiresolution B-splines. Advantage of this algorithm is that for calculation universal spline handling metric can be used, but disadvantage – there is no optimal algorithm.

## **Žmogaus judėjimo metodų kompiuterinėje animacijoje tyrimas**

### **SANTRAUKA**

Personažų animacija yra sudėtingiausia, bet tuo pačiu svarbiausia kompiuterinės animacijos sritis. Pagrindinė problema yra susijusi su realistiniu, lanksčiu žmogaus judėjimo atvaizdavimu, esant sudėtingiems bei gausiems skaičiavimams.

Hibridinis, žmogaus judėjimą analizuojantis algoritmas yra pristatomas teoriniu atžvilgiu. Šis algoritmas remiasi apribojimų įvedimu, kurių vienas iš pagrindinių yra laikomas žmogaus kūno dalių masės pasiskirstymas, konstruojami tarpiniai judėjimo vaizdai, kurių apjungimas yra apskaičiuojamas, panaudojant daugialygio B-splaino aproksimaciją. Šio algoritmo privalumas, kad skaičiavimams yra panaudojama universali splaino panaudojimo metrika, bet kaip trūkumą galima paminėti, kad tai nėra optimalus siūlomas algoritmas.

# TURINYS

ĮVADAS .....	8
1. ŽMOGAUS JUDĖJIMO METODŲ APŽVALGA .....	10
2. JUDĖJIMO TRANSFORMAVIMO ALGORITMAS, PAGRĮSTAS FIZIKINIAIS DĖSNIAIS .....	13
2.1. Judėjimo metodo transformacijos procesas .....	14
2.1.1. Personažo supaprastinimas .....	14
2.1.2. Įvykių erdvės priderinimas .....	16
2.1.3. Įvykių erdvės redagavimas .....	17
2.1.4. Judėjimo pertvarkymas .....	18
2.1.4. Mažiausios masės perkėlimo apskaičiavimas .....	19
2.2. Rezultatai .....	20
3. SAŲVEIKAUJANČIO JUDĖJIMO REDAGAVIMAS HIERARCHINIU BŪDU .....	22
3.1. Perkėlimo žymėjimas .....	22
3.2. Daugialygio B-splaino aproksimavimas .....	23
3.3. Hierarchinis judėjimo redagavimas .....	25
3.3.1. Apribojimai .....	26
3.3.2. Judėjimo derinimas .....	26
3.3.3. Mazgo išretinimas .....	27
3.3.4. Pradiniai spėjimai .....	28
3.4. Teorinės dalies išvados .....	30
4. HIBRIDINIS ALGORITMAS .....	31
4.1. Apribojimų nustatymas .....	31
4.2. Panašių reikšmių apskaičiavimas .....	33
4.3. Pereinamų taškų parinkimas .....	34
4.4. Perėjimo konstravimas interpoliuojant .....	36
4.5. Perėjimo konstravimas aproksimuojant B-splainais .....	38
4.6. Galutinio judėjimo generavimas .....	38
4.7. Realizavimas .....	38
IŠVADOS .....	39
LITERATŪRA .....	41
SANTRUMPŲ IR TERMINŲ ŽODYNAS .....	44



## Lentelių sąrašas

1 lentelė Apribojimų suderinamos poros .....	34
2 lentelė Bendrinės konstantos bei jų reikšmės.....	45

## Paveikslėlių sąrašas

1 pav. Karkasinės kaimynystės naudojimas, norint apibrėžti interpoliuojamas formas .....	11
2 pav. Algoritmo schema .....	14
3 pav. Kinematinis personažo suprastinimas .....	15
4 pav. Žmogaus bėgimui ir plačiam šuoliui pilnas ir supaprastintas personažų modeliai.....	15
5 pav. Raktiniai kadrai einant, sukryžiuojant kojas, šlubuojant, bėgant plačiu žingsniu .....	21
6 pav. Hierarchinės kreivės pritaikymas išsibarsčiusiems duomenims daugialygio B-splaino aproksimavime .....	24
7 pav. Mazgų išretinimas, gaunant platesnį deformacijos diapazoną .....	28
8 pav. Personažo judėjimas lygiu ir gruoblėtu paviršiumi .....	29
9 pav. Svorinių taškų aibė personažo griaučiuose .....	33
10 pav. Žmogaus judėjimo vaizdo atrinkimas, pasirenkant kaimyninius kadrus .....	36
11 pav. Ribinių kadru suvienodinimas ir pritaikymas .....	37

## IVADAS

Per keletą paskutiniųjų dešimtmečių animacijos sritis išsivystė taip, kad tapo pagrindiniu objektų vaizdavimo įrankiu televizijoje (filmuose, reklamose) bei kompiuteriuose (įvairiose projektavimo bei modeliavimo, mokomosiose programose, žaidimuose). Tam, kad tradicinės ir kompiuterinės animacijos sritys vystytųsi, buvo atliekami įvairūs moksliniai tyrimai, kad patobulintų šioje srityje dirbančių specialistų darbą. Tad tobulėjant kompiuteriams, programinei įrangai, plėtėsi ne tik kompiuterinės animacijos įvairios sritys, buvo stengiamasi daugiau objektų ir personažų veiksmų automatizuoti. Vaizdavimo procesas tapo sudėtingesnis, labai stengiantis, kad objektų judėjimas būtų kuo realistiškesnis.

N. M. Thalmann ir D. Thalmann [26] pasiūlė naują kompiuterinės animacijos vaizdų klasifikavimą, įtraukiant lanksčius kūnus (virtualūs gyvūnai, žmonės), atsižvelgiant tiek į judėjimo metodų valdymą, tiek pagal tarpinių veiksmų rūšį, kuriuos šie kūnai atlieka. Taigi personažų judėjimo valdymas yra skirstomas į tris kategorijas: geometrinį, fizikinį ir elgsenos. Pagrindinė problema personažų judėjimo valdyme yra pats judėjimo valdymo procesas, gaunamas imituojant fizikinius dėsnius, kurie valdo judėjimą realiame pasaulyje bei gana sudėtingi procesai. Kinematika grįstos sistemos yra daugiausiai intuityvios ir stokoja dinamiško integralumo, gaunami vaizdai yra trūkčiojantys arba atrodantys nenatūraliai.

Personažų animacija yra viena iš sudėtingiausių kompiuterinės animacijos sritis, reikalaujanti daugiausia žinių įvairiose srityse ir meistriškumo. Tačiau ši sritis yra taip pat labiausiai kritikuojama visuomenės. Šioje srityje viena iš pagrindinių problemų, susijusių su kompiuterine animacija, yra nerealistiškas personažų judėjimas, kuris vaizduojamas, atsižvelgiant į personažo fizikines savybes (dydis, kūno masė, judėjimo greitis, lankstumas ir t.t.), įvairius aplinkos apribojimus, pasitaikančias kliūtis, aplinkos savybes.

Kadangi kompiuterinėje animacijoje dažniausiai vaizduojamas veikėjas yra žmogus, tai nagrinėsime metodus ir algoritmus, kurių pagalba stengsimės atvaizduoti kuo tikslesnį, efektyvesnį, lankstesnį ir realistiškesnį judantį žmogų.

Vienas iš pasiūlymų ateičiai [25], siekiant gauti realistinį judėjimą, tuo pačiu panaudojant animatoriaus minimalias darbo pastangas, yra panaudoti tikslų, generuojamą atvirkštinės kinematikos pagalba judėjimą. Naudojant šį pasiūlymą animatorius neturi valdyti kiekvieno



personažo kūno dalies struktūroje tam, kad išlaikytų tam tikrą pozą. Pavyzdžiui, norint, kad figūra judėtų, viskas, ką animatorius turi padaryti yra nutempti žymeklį į vietą ir likusi kūno dalis seks paskui. Galbūt visa tai yra truputi pernelyg supaprastinama, bet tai tarnauja iliustruojant tikslo tiesioginės animacijos ketinimus.

Darbe pateikiama kompiuterinės animacijos žmogaus judėjimo metodų apžvalga, nagrinėjami algoritmai, vaizdavimo lygtys. Aptartos pagrindinės problemos.

Tiriamajoje dalyje aprašomas siūlomas hibridinio algoritmo, teorinis modelis, pristatant jo idėją, algoritmo aprašymas, realizacijos detalės.

Darbas baigiamas išvadomis.

## 1. ŽMOGAUS JUDĖJIMO METODŲ APŽVALGA

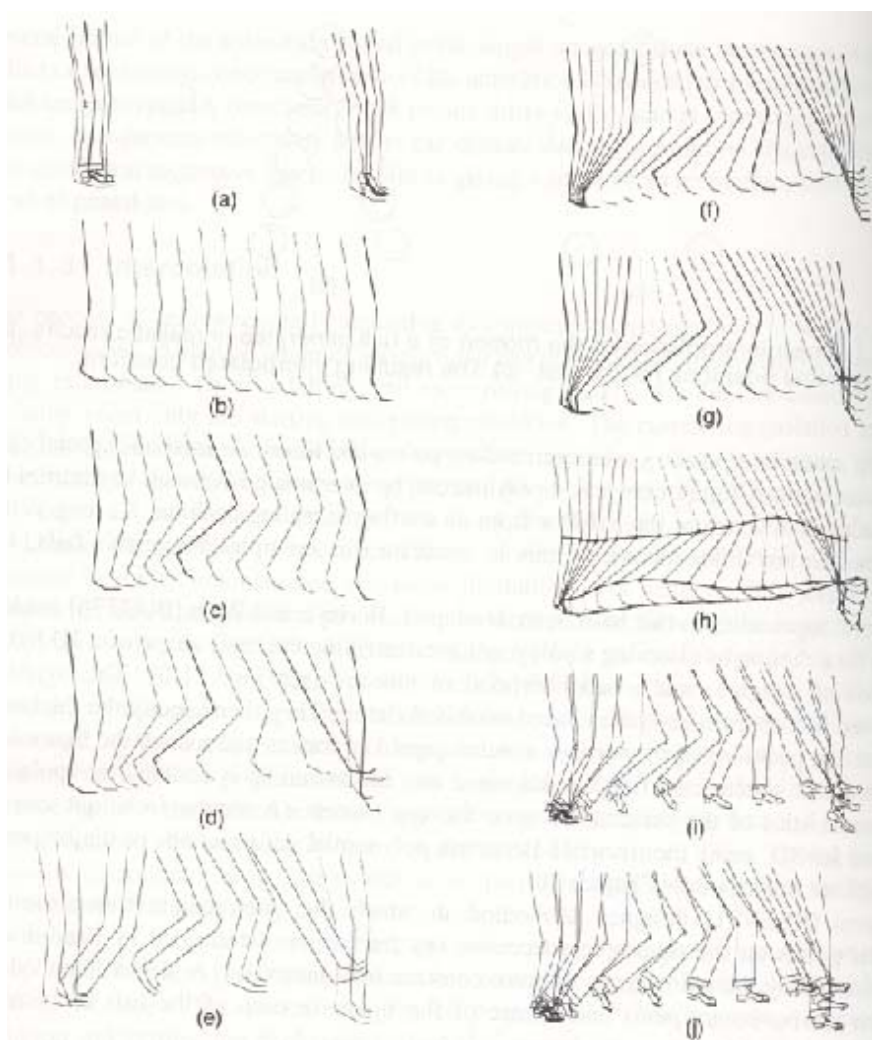
Pasirinkimas metodų ir algoritmų kompiuterinės grafikos objektų, personažų bei vaizdų modeliavimui yra šiuo metu pagrindinis kompiuterinės animacijos tikslas. Pagrindiniai kompiuterinės animacijos judėjimo metodai yra skirstomi į dvi kategorijas [19]:

1. **interpoliacija ir pagrindiniai metodai;**
2. **pažengusieji algoritmai.**

Šios dvi kategorijos buvo išrinktos specialiai tam, kad būtų galima žymėti skaičiavimų skirtumus tarp priėjimo prie judėjimo valdymo. Pirmoji grupė – žemesnio lygio metodai, kai animatorius tiksliai valdo judėjimą ir jo laukiamas rezultatas yra iš anksto įsivaizduojamas. Antroji grupė jungia aukštesnio lygio algoritmus, kuriuose valdymas vyksta ne taip detalai, mažiau įsivaizduojant, kaip tiksliai tas judesys atrodys.

Interpoliacija ir pagrindiniai metodai: kelio sekimas (angl. path following), *WYSIWYG* (angl. what you see is what you get) metodas [13], fizikiniu/biologiniu modeliais paremtas animacijų generavimas (angl. physically based animation), animavimo kalbos [19], objektų deformacija, morfinimas [24], įvairios išvaizdos detalių vaizdavimas (raumenų judėjimo atvaizdavimas, veido išraiškų simuliacija, plaukų judėjimo modeliavimas, ...) susideda iš metodų, kuriuose kompiuteris yra naudojamas detaliems skaičiavimams judėjimui perteikti, nurodžius animatoriui reikalingą informaciją. Naudojant šiuos metodus, animatorius pakankamai tiksliai įsivaizduoja norimą sukurti judėjimo vaizdą. Šie metodai leidžia animatoriams nurodyti tikslų uždavinį ir tiksliai valdyti judėjimą. Taigi atliekami atitinkamų skaitinių reikšmių skaičiavimai, naudojant kompiuterio funkcijas, kurios reikalingos, norint gauti norimą judesį.

Keletas pirmųjų žmogaus judėjimo metodų, naudojant interpoliaciją, buvo pristatyti Burtynk ir Wein [4] pagalba, kurie sukūrė žmogaus judėjimo schemą: pasirinktą daugiakampio kreivę aprašę esmine 2D figūros formos interpoliuojama funkcija, panaudojant raktinius kadrus. Šios kreivės tarpinės pozicijos, vadinamos karkasine kaimynyste, pavaizduotos 1 pav.



**1 pav.** Karkasinės kaimynystės naudojimas,  
norint apibrėžti interpoliuojamas formas (Burtnyk; Wein, 1976)

Antros kategorijos, pažengusieji algoritmai yra: automatinis kameros valdymas, hierarchinis kinematinis modeliavimas [19], atvirkštinė kinematika [13], kietų kūnų modeliavimas [24], t.y. modelių elgsena ir aukštesnio lygio procedūriniai algoritmai, kurie generuoja judėjimą, naudojant taisyklių (dėsningumų) arba apribojimų rinkinį, kurie nurodo, kas privalo būti padaryta vietoj nurodyto judėjimo ir, kaip tai turi būti padaryta. Animatorius pasirenka tinkamą algoritmą arba nustato reikalavimus modeliui, paskui išsirenka pradines reikšmes arba apribojimų sąlygas. Toliau, sistema pradeda vykdyti nurodymus ir objektų judėjimas automatiškai kuriamas algoritmu arba modeliu. Pvz. toks pritaikymas, kaip dinamika kietajam kūnui, dažnai remiasi sudėtingais skaičiavimais.

Praktikoje dauguma metodų ir algoritmų, naudojamų kompiuterinėje animacijoje, yra neatskiriami nuo žemiausio iki aukščiausio lygio abstrakcijos. Kiekvienas metodas reikalauja atitinkamo kiekio pastangų iš animatoriaus ir iš kompiuterio. Žemesnio lygio abstrakcijos reikalauja didesnio skaičiaus duomenų iš vartotojo ir naudoja pakankamai tiesinius skaičiavimus. Antra vertus aukštesnio lygio algoritmams reikia mažiau konkrečios informacijos iš animatoriaus ir daugiau skaičiavimų.

Bendru atveju paprasčiausios interpoliacijos metodai apibrėžia nurodytų taškų ar jų reikšmių skaičiavimus, kai taškai yra žinomi arba išdėstymą, naudojantis aplinkiniais taškais arba jų reikšmėmis.

Polilinijos (splainai) yra dažnai naudojamos kreivių vaizdavimui, dėl jų konstrukcijos paprastumo, jų paprastais ir tiksliais vertinimais, jų galimybėmis aproksimuoti sudėtingas figūras, pritaikant kreivę, ir sąveikaujant kreivės projektavimą

B-splainas yra splaino funkcija, kuri turi mažiausią palaikymą atžvilgiu duoto laipsnio, glotnumo ir srities padalijimo. B-splainas kompiuterinėje grafikoje dažnai remiasi splaino kreive, parametrizuota splaino funkcijomis, kurios yra išreikštos B-splainų tiesinėmis kombinacijomis.

Todėl dėl piešimo trūkumų buvo pradėta naudotis splainais vietoj pašalinimo interpoliacijos tarpiniuose pagrindiniuose kadruose. Splainai gali būti panaudojami įvairiems bet kuriems parametrų sulyginti. Splainų pagalba galima padaryti, kad individualūs taškai (individualūs objektai) judėtų sklandžiai erdvėje ir laike. Tačiau tai realizuojant iškyla problemos, kurias siūloma specifiškai tik kelių taškų pozicijomis, nagrinėjant keliais laiko momentais, ir išplėsti splaino interpoliaciją tarpiniuose taškuose bei laike.

Šiame darbe apžvelgiami du skirtingų kategorijų kompiuterinės animacijos žmogaus judėjimo metodai. Jie buvo pasirinkti, nes yra lengviausiai realizuojami, pirmu atveju siūlomas metodas padeda išlaikyti judėjimo vientisumą ir suteikia intuityvų judėjimo valdymą, esant nesudėtingiems skaičiavimams, bei gaunamas žmogaus judėjimo vaizdas atrodo gana realistiškas. Šių metodų pagrindu buvo vystomi ir tobulinami tolimesni 3D žmogaus judėjimo metodai. Pirmasis metodas – fizikiniais dėsniais paremta judėjimo transformacija ir antrasis, hierarchinis būdas, interaktyvaus judėjimo redagavimas.

## **2. JUDĖJIMO TRANSFORMAVIMO ALGORITMAS, PAGRĪSTAS FIZIKINIAIS DĖSNIAIS**

Vienas iš būdų sugeneruoti realistinį judėjimą, yra jį įgyti iš realaus pasaulio. Judėjimo užfiksavimo sistemos naudoja jutiklius, esančius ant judesį imituojančio žmogaus, tam, kad įrašytų absoliučias pagrindinių raktinių taškų padėtis ant personažo kūno per tam tikrą laiką ir gautų realistiškai atrodantį judėjimą. Tačiau nors šis animacijos tipas yra pakankamai realistiškas, jis užleidžia vietą nestruktūrizuotam ir tarpusavyje nesusijusiam judėjimui. Alternatyvus požiūris į šią problemą, realistiškas judėjimo sekos redagavimas, spręsti yra išgauti fizikinį modelį ir užfiksuoti duomenis, atlikti visą redagavimą kompiuterinio modelio.

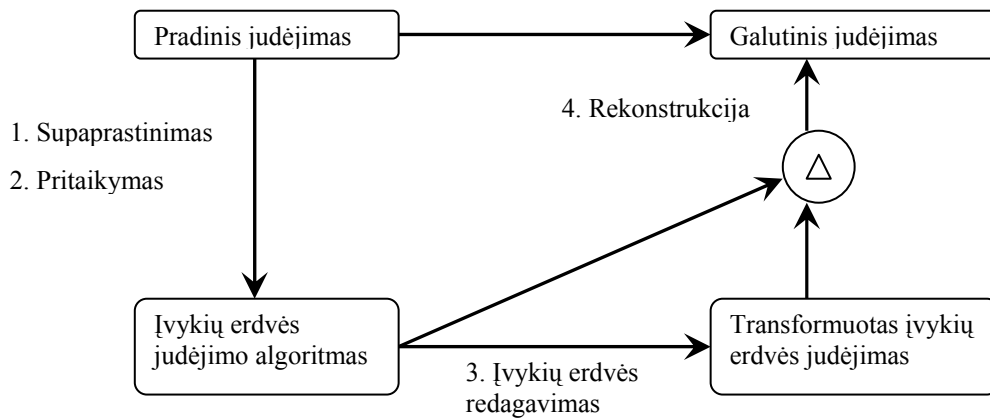
Šiuolaikiniai dinaminiai metodai skaičiuoja objektų, kurie paklūsta fizikiniams dėsniams, judėjimą. Aktyvus veikėjas sukuria judėjimą, pasinaudodamas savo raumenimis. Taigi specifinis realios būtybės judėjimas priklauso nuo jos sudėtingos raumenų griaučių struktūros. Nurodyti tiksliai raumenų jėgas, kurios padės animacijai atrodyti realistiškai, yra ypač sudėtinga. Dinaminuose metoduose kiekvienas animacijos kadras priklauso nuo prieš tai buvusio kadro. Mažiausi dinaminių savybių pokyčiai bet kuriame viename kadre įtakoja visai nuosekliai kadrų sekai, o rezultate gaunamas valdymo galimybės trūkumas.

Vartotojo nurodyti judėjimo apribojimai, pradinė objektų konfigūracija, nurodytos fizikinės savybės, visa tai vadinsime tiesiog įvykių erdvės apribojimais. Taigi įvykių erdvės apribojimų būdas pateikia efektyvų sprendimą tiek realistiško vaizdavimo, tiek personažo judėjimo valdymo galimybės nepritekliui [28, 5, 30, 21]. Įvykių erdvės struktūroje vartotojas pirmiausia patikslina padėties apribojimus, kurie privalo atitikti rezultatyvią judėjimo seką. Papildomai šiems apribojimams, vartotojas taip pat specifikuoja tikslo funkciją, kuri yra atlikimo metrika bei bendras visų personažo raumenų energijos sunaudojimas. Algoritmas apima šią įvykių erdvės specifikaciją ir nustato judėjimo trajektorijas, kurios sumažina tikslo funkciją, kai yra nurodyti apribojimai.

Dinamiška įvykių erdvės formuluotė leidžia redaguoti judėjimą intuityviai, vartojant aukšto lygio judėjimo sąvokas, tokias kaip laikas, pėdsakų patalpinimas, atstumo ir įvairių galūnių masė, kūno junginių kiekis ir gravitacija. Naudojantis įvykių erdvės apribojimų dinamine formuluote, personažų animacijos sekų transformavimu, algoritmas palaiko pradinio judėjimo sekos realizaciją, nesant nuostoliams, bei visiškai valdo vartotojo redaguojamą procesą.

## 2.1. Judėjimo metodo transformacijos procesas

Pirmiausia sukonstruojamas supaprastintas personažo (žmogaus) modelis ir supaprastintas modelio judėjimas pritaikomas užfiksuotuose judėjimo duomenyse. Šiam parengtam judėjimui yra išgaunamas fizikinis įvykių erdvės optimizavimo sprendimas, kuris įtraukia kūno masės savybes, padėtį, pėdsakų konstrukcijas, raumenis ir tikslo funkciją. Norint pakoreguoti animaciją, modifikuojami apribojimai ir fizikiniai modelio parametrai bei kiti įvykių erdvės optimizavimo parametrai (galūnių geometrija, pėdsakų pozicija, tikslo funkcija, svoris). Pagal šį pakeistą įvykių erdvės parametrizavimą, apskaičiuojama transformuoto judėjimo seka. Galų gale yra pažymimi supaprastinto judėjimo modelio pokyčiai ant pradinio judėjimo, kad būtų galima sukurti galutinę animacijos seką.

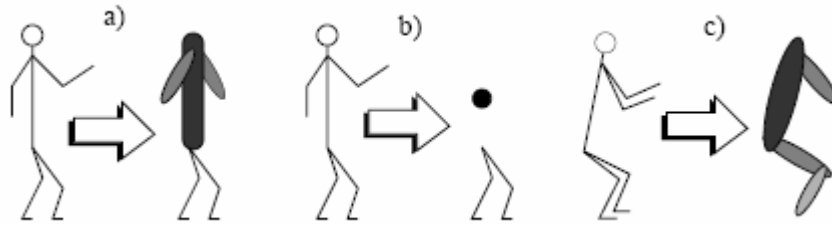


2pav. Algoritmo schema

Visas transformacijos procesas pasidalina į keturis pagrindinius etapus (2 pav.) [20]:

### 2.1.1. Personažo supaprastinimas

Sukuria abstraktų žmogaus modelį, talpinantį minimalų laisvumo laipsnių skaičių (Toliau DOF), reikalingą užfiksuoti įvedimo judėjimo esmei. Pažymimas įvedimo judėjimas ant supaprastinto modelio (3, 4 pav.).



**3 pav.** Kinematinis personažo supaprastinimas:  
*a) yra išskiriamos alkūnės ir stuburas; b) aukštesnioji kūno dalis yra sumažinama iki masės centro; c) simetrinio judėjimo gavimas (Popovic; Witkin, 1999)*

Vieną kartą supaprastinimus personažo modelį, pradinis judėjimas gali būti žymimas ant jo. Nuo tada, kai supaprastintas personažas turi daug mažiau DOFs, šis žymėjimas yra nutraukiamas. Dabar yra apibrėžiami „mygtukai“ (angl. handles) – daugiareikšmės, laike varijuojančios funkcijos, kurių pagalba animatorius kontroliuoja kūno dalių parametrus tiek sudėtingo, tiek supaprastinto personažo modelių atžvilgiu. Visi „mygtukai“ priklauso nuo personažo padėties, apibrėžtos reikšmių vektoriais kiekvienam DOF  $q(t_i)$ . Jie simbolizuoja intuityvius įvairių kūno savybių matmenis, tokias kaip 3D taško pozicijas, 3D kryptis, atstumas tarp dviejų nustatytų kūno taškų. Taigi pradinio judėjimo „mygtukų“ rinkinys, kaip  $h_o(q_o(t))$  (pradinis „mygtukų“ rinkinys) ir  $h_s(q_s(t))$  (galutinis „mygtukų“ rinkinys) tegu būna atitinkamai supaprastinto judėjimo „mygtukai“.



**4 pav.** Žmogaus bėgimui ir plačiam šuoliui visapusiškas ir supaprastintas personažų modeliai

Supaprastinto personažo judėjimas randamas, sprendžiant lygtį:

$$E_d = \left[ h_o \left( q_o(t_i) \right) - h_s \left( q_s(t_i) \right) \right]^2 \quad (1.1)$$

$$\min_{q_s(t_i)} E_d \quad (1.2)$$

kiekvienam karkasui (angl. frame)  $t_i$ . Šiame procese turėtų būti mažiausiai kaip daugelis „mygtukų“ taip čia esantis DOFs skaičius supaprastintame personaže. Tokiu būdu supaprastintas judėjimas apibrėžiamas  $h_s(q_s(t))$ .

### 2.1.2. Įvykių erdvės priderinimas

Tikslas - rasti įvykių erdvės optimizavimo problemą, kurios sprendimas artimai sutampa su supaprastinto personažo judėjimu.

Personažo kūno matmenų ir masės pasiskirstymas, naudojamas iš biomechaninių tyrimų šaltinių [7, 17].

Judantis personažas turi baigtinį kinematinį DOF ir raumenų skaičių. DOF yra vaizduojamas personažo galūnių sujungimuose, kai raumenys įtempia jėgas arba sąsukio momentus skirtingose kūno dalyse, taigi veikia judėjimą. Tiek kūno ir tiek raumenų DOF pokytį per laiką, bendrai žymėsime  $q(t)$  arba atskirai, kinematiškai  $q_k(t)$ , raumuo  $q_m(t)$ . Judėjimo užduotis yra rasti trokštamą žmogaus judėjimą, kuris atitiktų reikalavimų aibę (apribojimai, išorinės jėgos, tikslo funkcija). Asmens judėjimas remiasi padėties apribojimais ( $C_p$ ), papildomai aplinkai įvedus mechaninius apribojimus ( $C_m$ ) kiekį ant kūno. Pvz.: tam, kad priversti žmogaus poziciją išsitiesti, reikia abi pėdas sukonstruoti ant grindų. Svarbu užtikrinti, kad pėdos niekad neprasiskverbs pro grindų paviršių.

Toliau reikia užtikrinti dinamišką judėjimo korektiškumą. Tai padaroma apribojant kiekvieno DOF greitėjimą. Toliau yra užtikrinama, kad  $F=ma$  palaikytų visus laisvės laipsnius per tam tikrą laiką. Tokie apribojimai yra vadinami dinaminiais apribojimais ( $C_d$ ). Kai šie apribojimai yra patenkinami, tuomet rezultatyvus judėjimas yra fiziškai įmanomas ir duodama raumenims galimybė sugeneruoti jėgas.

Kai judėjimas yra apibrėžtas, optimizuojame tikslo funkciją  $E(q(t),t)$ , parametrizuodami erdvėje ir laike, priklausančią nuo padėties, mechaninių ir dinaminių apribojimų:

$$\min_{q(t)} E(q(t),t) \quad \text{objektą } i \quad \begin{cases} C_p(q(t),t) = 0 \\ C_m(q(t),t) = 0 \\ C_d(q(t),t) = 0 \end{cases} \quad (1.3)$$

Šiai funkcijai apskaičiuoti, siekiant išspręsti įvykių erdvės optimizavimo problemą, yra naudojamas retas SQP [9] metodas. Tai tolydus optimizavimo metodas, kuris reikalauja visų apribojimų ir tikslo funkcijų pirmų išvestinių apskaičiavimo.

Raumenys yra pagrindinis personažo judėjimo šaltinis. Todėl jiems pavaizduoti yra naudojamos apibendrintos raumenų jėgos  $Q$ . Apibendrinto raumens buvimas kiekviename personažo DOF, vaizduoja minimalią raumenų aibę, kuri užtikrina visapusišką personažo



judėjimo diapazoną. Kiekvienas kinematinis DOF  $q_{k_i}$  turi atitinkamai prislopintą generalizuotą raumenų jėgą

$$\mathbf{Q}_{k_i} = \mathbf{k}_s (\mathbf{q}_{k_i} - \mathbf{q}_{m_i}) - \mathbf{k}_d (\dot{\mathbf{q}}_{k_i} - \dot{\mathbf{q}}_{m_i}) \quad (1.4)$$

kur  $q_{m_i}$  yra papildomas raumuo DOF, dažnai interpretuojamas kaip reikalaujama  $q_{k_i}$  reikšmė.

Dauguma padėties ir mechaninių apribojimų iškreinta, įvedus judėjimo charakteristiką. Pvz.: bėgimo ar ėjimo sekoje yra tikslinami mechaniškų taškų apribojimai kiekviename periode, kai kojos pėda liečiasi su grindimis. Taip yra išvengiama šalutinių apribojimų.

Remiantis judėjimo optimalumo tyrimais [2, 18] ir norint išvengti spėliojimų nustatant teisingą tikslo funkciją, laikysime, kad prasidedantis judėjimas yra labai artimas optimaliam. Pirmiausia yra išmatuojamas nukrypimas nuo originalaus judėjimo ( $E_d$ ), kaip aprašyta 1.1 formulėj. Toliau yra įtraukiama raumenų glotnumo tikslo komponentė  $E_m = \dot{q}_m^2$ . Įvykių erdvės tikslas yra svorinė dviejų komponentių suma:

$$\mathbf{E} = w_d \int \mathbf{E}_d + \int \mathbf{E}_m \quad (1.5)$$

Tuomet, kai Niutono apribojimo liekanai tampa labai maža, palaiapsniui mažinama  $w_d$  nuo pradžios iki galo į 0. Iš anksto  $E_d$  komponentės egzistavimas optimizavimo procese neleidžia optimizavimo metodui nukrypti nuo pirminio judėjimo tol, kol įvykių erdvės apribojimai yra patenkinami.

Po konvergavimo yra užbaigiama įvykių erdvės problemos apibrėžimas, kurios sprendinys yra artimas pradiniam judėjimui. Buvo sukonstruotas intuityvus įvykių erdvės apribojimų „valdymo mygtukas“, kuris gali būti redaguojamas tam, kad sukurti artimą, neišsenkanti, skirtingą realistiško judėjimo sekų kiekį.

### 2.1.3. Įvykių erdvės redagavimas

Tikslas – įvykių erdvės judėjimo parametrų keitimas sistemoje, teikiamos naujos apribojimų padėtys, keičiama personažų kinematika, tikslo funkcija ir t.t.

Įvykių erdvės apribojimų parametrizavimas numato galingą ir intuityvų dinaminės animacijos daugumos aspektų valdymą. Keičiantis egzistuojantiems apribojimams, vartotojas gali pakeisti pėdos išsidėstymą tiek laike, tiek erdvėje. Pvz.: žmogaus bėgimo seka gali būti pakeista į zigzaginį bėgimą.

Pokyčiai gali atsirasti, keičiant personažo modelį (galūnių matmenis, jų masės pasiskirstymo charakteristikas) ir taip bus pastebėtas dinamiškas judėjimo pokytis. Galima netgi pašalinti kūno dalis, apriboti DOF iki specifinio rinkinio, arba pašalinti DOF visiškai, efektyviai patalpinant žinomas kūno dalis į judėjimo vaidmenį (šlubuojančio žmogaus bėgimas). Judėjimas gali būti taip pakeistas: pridedant papildomas atitinkamas svorines tikslo komponentes (minkščiau atrodantis bėgimas, gaunamas pridedant tikslo komponentę, kuri minimizuoja grindis veikiančias jėgas arba atitinkamai „kietas“ bėgimas). Po kiekvieno redagavimo, yra iš naujo persprendžiamas įvykių erdvės optimizavimas ir kuriama nauja transformuota animacija. Kol optimizavimo pradinis taškas yra arti reikalaujamo sprendimo, visi dinaminiai apribojimai yra patenkinami pradžioje, tada optimizavimas konverguoja sparčiai.

#### 2.1.4. Judėjimo pertvarkymas

Pokyčio peržymėjimas judėjime pristatomas įvykių erdvės redagavimu ant pradinio judėjimo. Gaunama galutinė animacija.

Tam, kad būtų padaryta visapusiška personažo modelio transformuota animacija, yra perkonstruojama galutinis judėjimas nuo pradinio judėjimo ir du supaprastinti įvykių erdvės judėjimai. Pritaikius transformaciją, pradinei sekai, animatorius modifikuoja fundamentalias, dinamines judėjimo savybės tol, kol originale yra išsaugomos specifiniai sudėtingi elementai. Pertvarkymas priklauso tiek nuo įvykių erdvės apribojimų, tiek nuo judėjimo. Visi įvykių erdvės apribojimai yra sužymimi į jų pilnus personažų atitikmenis. Turint baigtinį įvykių erdvės redagavimo etapą, įgyjame tris skirtingas „mygtukų“ aibes:

- Pradinio judėjimo „mygtukai“  $h_0(q_0)$ .
- Įvykių erdvės derinimo „mygtukai“  $h_s(q_s)$ .
- Transformuotos įvykių erdvės „mygtukai“  $h_t(q_t)$ .

Galutinio judėjimo „mygtukai“ yra apibrėžiami kaip

$$\mathbf{h}_f(\mathbf{q}_f) = \mathbf{h}_0(\mathbf{q}_0) + (\mathbf{h}_t(\mathbf{q}_t) - \mathbf{h}_s(\mathbf{q}_s)), \quad (1.6)$$

perkeliant pradinius „mygtukus“ pagal skirtumus tarp dviejų įvykių erdvės sprendimų. Kai dešinioji formulės 1.6 pusė yra žinoma, atrodys, kad sprendžiant atvirkštinės kinematikos problemą, randant  $\mathbf{q}_f$  galėtų užbaigti pertvarkymą. Tačiau, „mygtukų“ skaičius yra daug mažesnis nei DOFs skaičius pilname personaže, taigi iškyla problema, kuri neleidžia tiesiogiai rasti  $\mathbf{q}_f$  be sudarymo papildomų DOFs.

Sudarant papildomus DOFs, bandoma tiesiogiai rasti  $q_f$  pagal 1.7 formulę:

$$\min_{q_f} E_{dm}(q_0, q_f) \text{ objektą i } \begin{cases} C(q) = 0 \\ h_f(q_f) = h_0(q_0) + (h_t(q_t) - h_s(q_s)) \end{cases} \quad (1.7)$$

Toliau pirmiausia yra formuluojama originaliam judėjimo glaudumo matmuo. Paprasta tikslo funkcija, kuri matuoja kiekvieno DOF nukrypimą  $E_{ad} = (q_f - q_0)^2$ , įgauna nepageidaujamus rezultatus. Norint to išvengti, suprojektuojama naujas tikslas  $E_{dm}$ , kuris matuoja perkeliama masės išsidėstymą tarp dviejų padėčių.

#### 2.1.4. Mažiausios masės perkėlimo apskaičiavimas

Duotos dvi personažo padėties, aprašytos DOF reikšmėmis  $\bar{q}$  ir  $q$ . Skaičiuojama bendras perkėlimo masės  $E_{dm}(\bar{q}, q)$  kiekis, kai kūnas transformuojamas nuo  $\bar{q}$  padėties į  $q$  padėtį. Visa perkelta masė yra masės perkėlimų suma kiekvienam mazgui  $k$  hierarchijoje  $E_{dm} = \sum_k E_k$ .

Toliau yra skaičiuojamas mazgo masės perkėlimas  $E_k$ , kaip kūno taško  $p_i$  perkėlimas mastelyje su jo mase  $\mu_i$  integruojamą per visus kūno mazgo taškus  $k$

$$E_k = \int \int \int_i \mu_i (p_i - \bar{p}_i)^2 dx dy dz, \quad (1.8)$$

kur kiekvienas “uždraustas“ simbolis remiasi kiekiu suskaičiuotu padėtyje  $\bar{q}$ .

Kadangi yra reikalingas santykinis masės perkėlimas, yra skaičiuojami kūno padėčių globalaus keitimosi vietomis pastovumas ir perkėlimas. Jei

$$p'_i = R_0 R_1 \dots R_{j-1} R_j x_i \quad (1.9)$$

yra pasaulinės erdvės kūno taško  $x_i$  padėtis personažo hierarchijoje mazge  $j$  ir perkėlimas  $R_0$  talpina globalų keitimąsi vietomis ir hierarchijos perkėlimą, yra apibrėžiama

$$p_i = R_1 \dots R_{j-1} R_j x_i = W_j x_i \quad (1.10)$$

Šis pažymėjimas leidžia supaprastinti  $E_k$

$$\begin{aligned} E_k &= \int \int \int_i \mu_i (W_i x_i W_i x_i - 2W_i x_i \bar{W}_i x_i + \bar{W}_i x_i \bar{W}_i x_i) dx dy dz \\ &= \text{tr} \left( W_i M_i W_i^T - 2W_i M_i \bar{W}_i^T + \bar{W}_i M_i \bar{W}_i^T \right) \\ &= \text{tr} \left( W_i M_i (W_i - 2\bar{W}_i)^T \right), \end{aligned} \quad (1.11)$$

kur  $tr()$  yra matricos užrašo operatorius ir mazgo  $i$   $M_i$  masės matricos tenzorius (angl. tenzor) yra apskaičiuojamas kaip integralas virš tolimesnės sandaugos kūno taškų  $x_j$ , sužymėtų mazgo masė  $m_i$

$$\mathbf{M}_i = m_i \int \int \int_j \mathbf{x}_j \mathbf{x}_j^T d\mathbf{x} d\mathbf{y} d\mathbf{z} \quad (1.12)$$

Kadangi  $\bar{W}_i M_i \bar{W}_i^T$  yra konstanta (nepriklauso nuo  $q$ ), tai ji yra pašalinama iš išsireiškimo.

Toliau yra apskaičiuojamos išvestinės atžvilgiu kinematinių DOFs

$$\begin{aligned} \frac{\partial E_k}{\partial q_j} &= tr \left( \frac{\partial W_i}{\partial q_j} M_i (W_i - 2\bar{W}_i)^T + W_j M_i \frac{\partial W_i^T}{\partial q_j} \right) \\ &= tr \left( 2 (W_i - \bar{W}_i) M_i \frac{\partial W_i^T}{\partial q_j} \right) \end{aligned} \quad (1.13)$$

Tiek  $E_k$  ir tiek  $\frac{\partial E_k}{\partial q_j}$  gali būti efektyviai apskaičiuojamos, naudojant rekursinius skaičiavimus

kiekvieno mazgo tarpinėms išraiškoms  $M_i W_i^T$  ir  $M_i \frac{\partial W_i^T}{\partial q_j}$ .

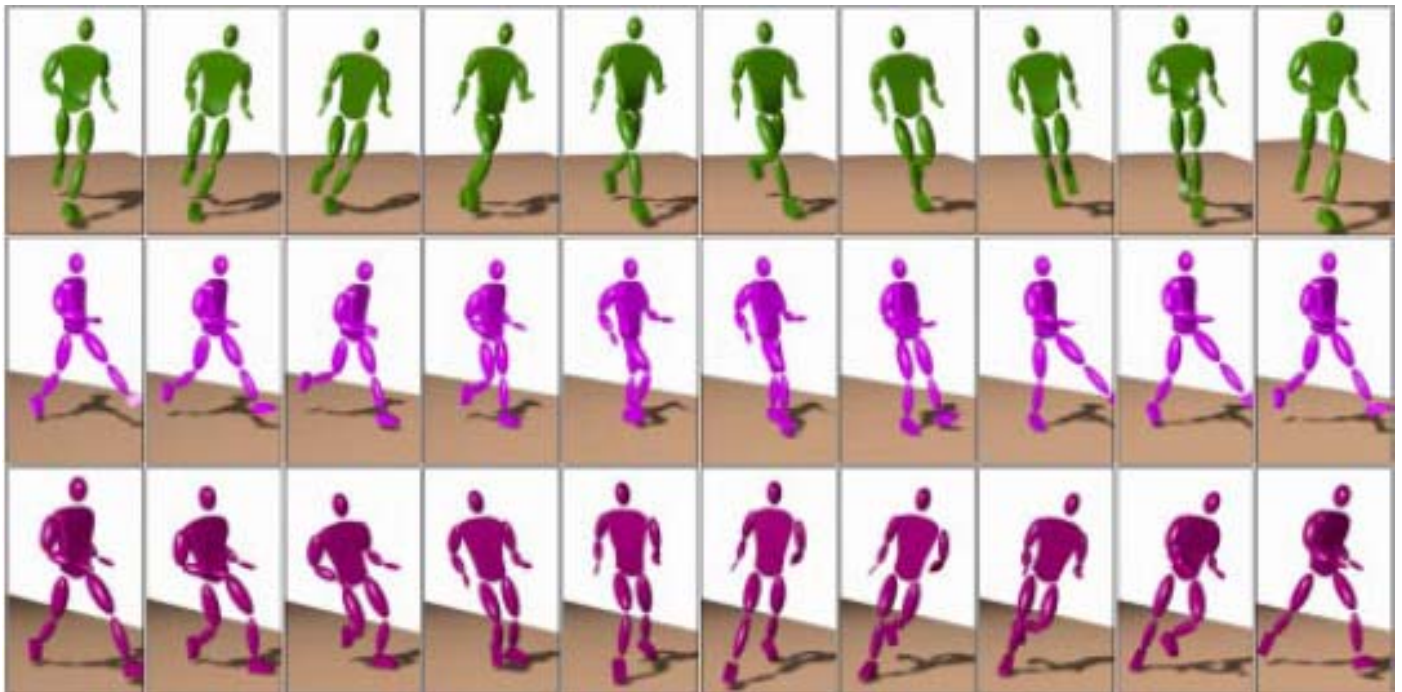
Nuo tada, kai rekonstrukcijos procesas yra atliekamas atskirai kiekvienam kadru, judėjimas rezultate gali pasirodyti neglotnus. Tuomet apibrėžiant animacijos intervalus, kuriuose yra reikalaujamas patobulintas glotnumas.

## 2.2. Rezultatai

Apibendrinant algoritmo proceso etapus, buvo galima pastebėti, kad personažo supaprastinimas ir įvykių erdvės derinimo etapai reikalauja iš animatoriaus daug pastangų. Vieną kartą apskaičiavus erdvinį modelį, jis gali būti pakartotas, kad sugeneruotų platų skirtingą animacijos asortimentų diapazoną. Įvykių erdvės redagavimas ir judėjimo rekonstrukcijos etapai yra visiškai automatizuoti. Šie etapai trunka daug mažiau laiko, vykdant skaičiavimus, negu pirmieji du etapai, kurie leidžia apskaičiuoti transformuotos judėjimo seką tarp artimai sąveikaujančių greičių.

Šiame metode buvo panaudoti labai detalaus judėjimo užfiksavimo duomenys kaip įvedimo sekos. Derinimo etape buvo panaudoti labai radikalūs supaprastinimo būdai tam, kad parodytų algoritmo lankstumą. Taip pat šiame metode reikalaujamas minimalus intuityvių pokyčių kiekis kiekvienai transformuojamai judėjimo duomenų sekai.

Personažo judėjimo procesas yra pavaizduotas 5 pav.:



**5 pav.** Raktiniai kadrai einant, sukryžiuojant kojas, šlubuojant, bėgant plačiu žingsniu

Visas aprašytas judėjimo sekas būtų sunku sukurti su jau egzistuojančiais judėjimo redagavimo įrankiais. Kai kurioms realistiškoms judėjimo sekoms sugeneruoti, reikalaujama milžiniškas skaičiavimų kiekis.

### **3. SAŲEIKAUJANČIO JUDĖJIMO REDAGAVIMAS HIERARCHINIU BŪDU**

Pateikiamas žmogui būdingo judėjimo pritaikymo metodas, norint įgauti trokštamą savybę, specifikuojant apribojimų aibę. Metode yra suderinama hierarchinės kreivės atitinkama metodika su atvirkštinės kinematikos sprendiniu. Naudojant kinematinį sprendinį, galima koreguoti žmogaus formą, norint patenkinti apribojimus kiekviename kadre. Kiekvienos jungties judėjimo perkėlimas apribotuose kadre yra interpoliuojamas ir vientisai skleidžiamas į kadrus. Toliau galima pridėti pritaikant judėjimo detales, norint patenkinti apribojimus nurodytus leistinam nukrypimui, pritaikant daugialygio B splaino atvaizdavimą, kuris tuo pačiu interpoliacijai suteikia pagreitėjimą.

Šiame metode yra naudojami daugialygiai B-splainai – naudojama daug splainų su skirtingais atstumais išdėstytais mazgais. Šie daugialygiai B-splainai sukuria mazgo sekų iš grubaus į smulkų hierarchijos panaudojimą, kad sugeneruoti vienodų kubinių B-splaino kreivių rinkinius, kurios sumuoja reikalaujamas funkcijas. Kiekviename hierarchijos lygyje B-splaino kreivės valdymo taškai yra skaičiuojami mažiausių kvadratų metodu. Išnaudojant žinomas daugialygio B-splaino kreivės savybes yra atitinkamai išvedama perkėlimo žymėjimų hierarchija, kuri yra pritaikoma pradinio judėjimo duomenims, norint išgauti naują, glaudžiai modifikuotą judėjimą. Po šio perkėlimo sužymėjimo, pradinio judėjimo detalios savybės gali būti išsaugomos.

Šio algoritmo metodika susideda iš supažindinimo su judėjimo perkėlimo žymėjimu, daugialygio B-splaino suderinamumo būdas, toliau pristatomas hierarchinis judėjimo redagavimo būdas, aprašomas atvirkštinės kinematikos sprendinių panaudojimas ir gaunami rezultatai.

#### **3.1. Perkėlimo žymėjimas**

Žmogaus figūros forma yra specifikuojama jos jungties kampais, pirminio segmento padėties ir orientacijos atžvilgiu. Toliau yra nurodoma pirminio 3D vektoriaus pozicija ir kiti vienetiniai kvaternionai (nekomutatyvių kompleksinių skaičių išplėtimai). Naudojami vienetiniai kvaternionai gali atvaizduoti 3D orientaciją glaudžiai ir kompaktiškai be ypatumų [23]. Šis vaizdavimas gali taip pat aprašyti tinkamai žmogaus junginius.

Judėjimas yra laike kintanti funkcija, kuri parodo žmogaus formą tam tikru laiku. Pažymime judėjimą  $m(t) = (p(t), q^1(t), \dots, q^n(t))^T$ , kur  $p(t) \in R^3$  ir  $q^1(t) \in S^3$  aprašo perkeliama ir sukama pagrindinio segmento judėjimą, ir  $q^i(t) \in S^3$  duoda sukamąjį  $(i-1)$ -ojo junginio judėjimą  $2 \leq i \leq n$ .

Perkėlimo žymėjimas yra apibrėžiamas kaip  $d(t) = m(t) \Theta m_0(t)$ , kur  $d(t) = (v^0(t), \dots, v^n(t))^T$  ir  $v^i(t) \in R^3$ , kai  $0 \leq i \leq n$ . Taigi naujas judėjimas gali būti gaunamas taikant perkėlimo žymėjimą į pradinį judėjimą, kaip  $m(t) = m_0(t) \oplus d(t)$ . Tai yra

$$\begin{pmatrix} p \\ q^1 \\ \dots \\ q^n \end{pmatrix} = \begin{pmatrix} p \\ q_0^1 \\ \dots \\ q_0^n \end{pmatrix} \oplus \begin{pmatrix} v^0 \\ v^1 \\ \dots \\ v^n \end{pmatrix} = \begin{pmatrix} p_0 + v^0 \\ q_0^1 \exp(v^1) \\ \dots \\ q_0^n \exp(v^n) \end{pmatrix}. \quad (2.1)$$

Čia  $\exp(v)$  nurodo 3D pasukimą apie ašį  $\frac{v}{\|v\|} \in R^3$  kampu  $\|v\| \in R$ . Kartu su perkėlimo žymėjimu, galima nagrinėti tiek poziciją, tiek orientacijos duomenis vienodu būdu; perkėlimo žymėjimas yra homogeninio 3D vektoriaus rinkinys, kol žmogaus forma yra pavaizduojama įvairiarūšio vektoriaus rinkiniu ir vienetiniais kvaternionais.

### 3.2 Daugialygio B-splaino aproksimavimas

Daugialygio B-splaino aproksimavimo būdą, išsibarsčiusių duomenų taškų surinkimą į splaino paviršių, pasiūlė Lee [22]. Kadangi kreivę valdyti reikia labiau nei paviršių, didesnis dėmesys yra atkreipiamas į kreivės surinkimą.

Tegu  $\Omega = \{t \in R \mid 0 \leq t < n\}$  būna srities intervalas. Laikykime išsibarsčiusių duomenų taškų aibę  $P = \{(t_i, x_i)\} t_i \in \Omega$ . Norint interpoliuoti duomenų taškus, yra suformuluojama aproksimavimo funkcija  $f$  kaip B-splaino pagrindinė funkcija, kuri yra apibrėžiama vienodo mazgo seka, vyraujant sričiai  $\Omega$ . Funkcija  $f(t) = \sum_{k=0}^3 B_k(t - [t]) b_{[t]+k-1}$  gali būti aprašyta valdančiųjų taškų išraiškomis ir vienodomis kubinėmis B-splaino pagrindinėmis funkcijomis  $B_k$ ,  $0 \leq k \leq 3$ . Čia  $b_i$  yra  $i$ -tasis valdantysis taškas mazgo sekoje  $-1 \leq i \leq n+1$ . Šio formulavimo pagalba išvestos  $f$  funkcijos problema yra sumažinama, norint rasti valdančiuosius taškus, kurie geriausiai aproksimuoja duomenų taškus  $P$ .

Kadangi kiekvienas valdantysis taškas  $b_j$  yra įtakojamas šalia esančiais duomenų taškais, galima apibrėžti artimumo (angl. proximity) aibę  $P_j = \{(t_i, x_i)\} \in P \{j - 2 \leq tI \leq j + 2\}$ , kuri įtakoja  $b_j$  reikšmę. Tiesinė algebra, naudodama pseudo atvirkštinę tvarką numato mažiausią kvadrato sprendinį:

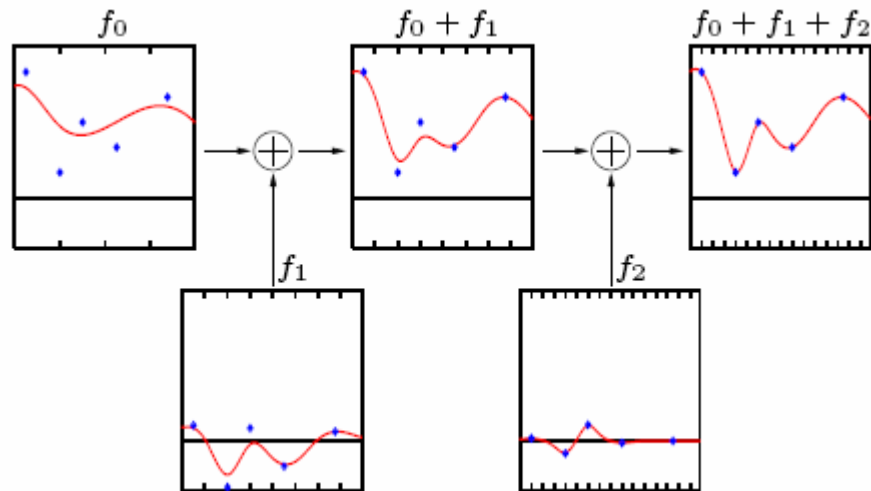
$$b_j = \frac{\sum_{(t_i, x_i) \in P_j} \omega_{ij}^2 \beta_{ij}}{\sum_{(t_i, x_i) \in P_j} \omega_{ij}^2} \quad (2.2)$$

kuris minimizuoja lokalinę aproksimavimo paklaidą  $\sum_{(t_i, x_i) \in P_j} \|f(t_i) - x_i\|^2$ . Čia

$\omega_{ij} = B_{j+1 - \lfloor t_i \rfloor}(t_i - \lfloor t_i \rfloor)$  gaunamas iš B-splaino pagrindinės funkcijos, ir

$\beta_{ij} = \frac{\omega_{ij} x_i}{\sum_{k=0}^3 B_k(t_i - \lfloor t_i \rfloor)}$  duoda efektą ištempti kreivę link duomenų taško  $(t_i, x_i)$ .

Čia yra atsižvelgiama tarp aproksimuojančios formos glaudumo ir tikslumo. Daugialygio B-splaino aproksimavimas naudoja B-splaino funkcijų eiles su skirtingu mazgo išdėstymu, norint įgyti glaudžią formą, kol tikslus aproksimavimas duoda duomenų taškus aibėje P. Funkcija nuo stambios mazgo sekos numato nelygų aproksimavimą, kuris toliau yra tobulinamas, tiksliai išgavus funkciją iš vėlesnio smulkaus mazgo sekų.



**6 pav.** Hierarchinės kreivės pritaikymas išsibarsčiusiems duomenims daugialygio B-splaino aproksimavime

Daugialygio B-splaino funkcija yra kubinio B-splaino funkcijų suma  $f_0, \dots, f_h$  (5 pav.), kuri yra apibrėžiama per vienodas mazgo sekas, esančias  $\Omega$  srityje. Tarkime, kad  $f_i$



mazgo išretinimas yra stambesnis nei  $f_j$ , bet kokiems  $i < j$ . Daugialygio B-splaino aproksimavimas prasideda, apibrėžiant  $f_0$  valdančiuosius taškus 1.15 formulėj.  $f_0$  gali palikti nukrypimus  $\Delta^1 x_i = x_i - f_0(t_i)$  kiekvienam taškui  $(t_i, x_i)$  aibėje P. Sekanti smulki funkcija  $f_1$  yra naudojama aproksimuoti skirtumą  $D_1 = \{(t_i, \Delta^1 x_i)\}$ . Tuomet suma  $f_0 + f_1$  nusileidžia mažesniai nukrypimui  $\Delta^2 x_i = x_i - f_0(t_i) - f_1(t_i)$  kiekvienam taškui  $(t_i, x_i)$ . Hierarchijos  $k$  lygyje, funkcija  $f_k$  yra gaunama aproksimuoti  $D_k = \{(t_i, \Delta^k x_i)\}$ , kur  $\Delta^k x_i = x_i - \sum_{l=0}^{k-1} f_l(t_i)$ . Kartojant šį procesą iki detaliu lygiu, galima padidintai įgyti baigtinę aproksimavimo funkciją  $f = \sum_{k=0}^h f_k$ .

### 3.3. Hierarchinis judėjimo redagavimas

Tarkim turime pradinį judėjimą  $m_0$  ir apribojimų aibę C, reikia išgauti tokį vientisą perkėlimo žymėjimą d, kad objekto judėjimas  $m = m_0 \oplus d$  atitiktų apribojimus C aibėje. Dabartinės judėjimo metodikos pavaizduoja perkėlimo žymėjimą kaip splaino kreivių aibę, kurios yra apibrėžtos bendra mazgo seka [11, 29]. Kiekviena splaino kreivė duoda kintančio laike judėjimo perkėlimą, atžvilgiu atitinkamo junginio. Smulkios mazgo sekos atžvilgiu, galima rasti sprendimą, kuris tiksliai atitiks visus apribojimus aibėje C.

Pritaikoma hierarchinė struktūra, apžvelgta 3.2 skyriuje. Nuo perkėlimo žymėjimo d, gaunama viena paskui kitą detalių perkėlimų dalies seka  $d_1, \dots, d_h$ , kuri duoda atitinkamai augančią patobulinto judėjimo eilę  $m_1, \dots, m_h$ .

$$m_h = \left( \dots \left( (m_0 \oplus d_1) \oplus d_2 \right) \oplus \dots \oplus d_h \right) \quad (2.3)$$

Čia  $d_k, 1 \leq k \leq h$  yra vaizduojamas kubinio B-splaino kreivių aibe 3D vektoriaus erdvėje.  $d_k$  komponentės kreivės yra apibrėžiamos bendra mazgų seka  $\tau_k$ , kurios yra vienodai išretinamos. Mazgo sekos  $\tau_k, 1 \leq k \leq h$ , formuoja iš grubaus į smulkų hierarchiją.  $\tau_1$  yra patalpinamas grubiame hierarchijos lygyje ir  $\tau_h$  - smulkesniai lygyje. Judėjimas  $m_1 = m_0 \oplus d_1$  numato apytikslį aproksimuotą objekto judėjimą, kuris toliau yra apibrėžiamas, pritaikant  $d_2$ , kad duotų daugiau tikslų aproksimavimą  $m_2$ . Pritaikant dalinį žymėjimą vienas prie vieno, galima padidinančiai išgauti galutinį judėjimą.

### 3.3.1. Apribojimai

Norint patikslinti reikalaujamo objekto judėjimo savybes, yra naudojamos dvi apribojimų kategorijos. Pirmoji kategorija yra naudojama aprašyti pačiam save (žmogui), kaip junginių ribojimas ir anatomicinis santykis junginiuose. Priklausantys kitai apribojimų kategorijai yra patalpinamos figūros galūnės tam tikrose padėtyse ir kryptyse, kurios yra sąveikaujančiai specifikuojamos vartotoju arba automatiškai išgaunamos iš sąveikos tarp figūros ir jos aplinkos. Pvz. pirmiausia specifikuojam kontaktinį tašką tarp pėdos ir pagrindo grafinėje sąsajoje ir automatiškai modifikuojame tašką, remiantis geometrinio pagrindo pokyčių kitimu. Taigi sutvarkyta pora  $(t_j, C_j)$  specifikuoja apribojimų aibę  $C_j$   $t_j$  kadre.

### 3.3.2. Judėjimo derinimas

Norint apskaičiuoti perkėlimo žymėjimą, yra svarbu įvertinti judėjimo kiekvieno junginio perkėlimą kiekviename apribotame kadre. Junginio perkėlimas tam tikrame kadre yra interpoliuojamas atitinkama komponento perkėlimo žymėjimo kreive ir todėl vientisai skleidžiamas į kaimyninius kadrus. Tarkim, kad esame  $k$ -ajame lygyje,  $1 \leq k \leq h$ . Kiekviename apribotame  $t_j$  kadre, atvirkštinės kinematikos sprendinys duoda personažo formą  $m^{t_j}$ , taip atlikus duodama apribojimų  $(t_j, C_j)$  aibė. Kadangi gali egzistuoti dauguma galimų formų, kurios atitinka visus apribojimus  $C_j$  aibėje, nuosekliai pasirenkame vieną, kuri minimaliai nukryptų nuo judėjimo  $m_{k-1}$  į ankstesnį lygį. Toliau yra minimizuojama

$$\|d^{t_j}\|_{\alpha}^2 = \|m^{t_j}(-)m_{k-1}(t_j)\|_{\alpha}^2 = \sum_{i=0}^n \alpha \|v^i\|^2 \quad (2.4)$$

kur  $d^{t_j} = (v^0, \dots, v^n)$ . Galima valdyti individualaus junginio tvirtumą, pakoreguojant jo svorinę reikšmę.

Jungiant atvirkštinės kinematikos sprendimą su hierarchine struktūra, duota 1.16 formulėje, yra nurodomas judėjimo derinimo algoritmas:

*Įvedimas: pradinis judėjimas  $m_0$ , apribojimų  $C$  aibė*

*Išvedimas: naujos judėjimo sekos  $m_h$*

1: For  $k:=1$  to  $h$  do

2:      $D:=0$

3:     Kiekvienam  $(t_j, C_j) \in C$

4:          $m^{t_j} := AK_{sprendinys}(C_j, m_{k-1}(t_j))$

5:          $d^{t_j} := m^{t_j}(-)m_{k-1}(t_j)$

- 6:  $D := D \cup (t_j, d^j)$   
 7: *Pabaiga for*  
 8: *Skaičiuoti  $d_k$  kreivę derinant į  $D$*   
 9:  $m_k := m_{k-1} \oplus d_k$   
 10: *Pabaiga for*

Šis algoritmas įvertina  $d_k$ ,  $1 \leq k \leq h$ , iš stambaus į smulkų tvarką. Kiekviename  $k$  lygyje hierarchijoje, yra skaičiuojamas judėjimo perkėlimas  $d^j := m^{t_j}(-) m_{k-1}(t_j)$  visiems  $(t_j, C_j)$  aibėje  $C$ , naudojant atvirkštinės kinematikos sprendinį (2-7 algoritmo eilutės).

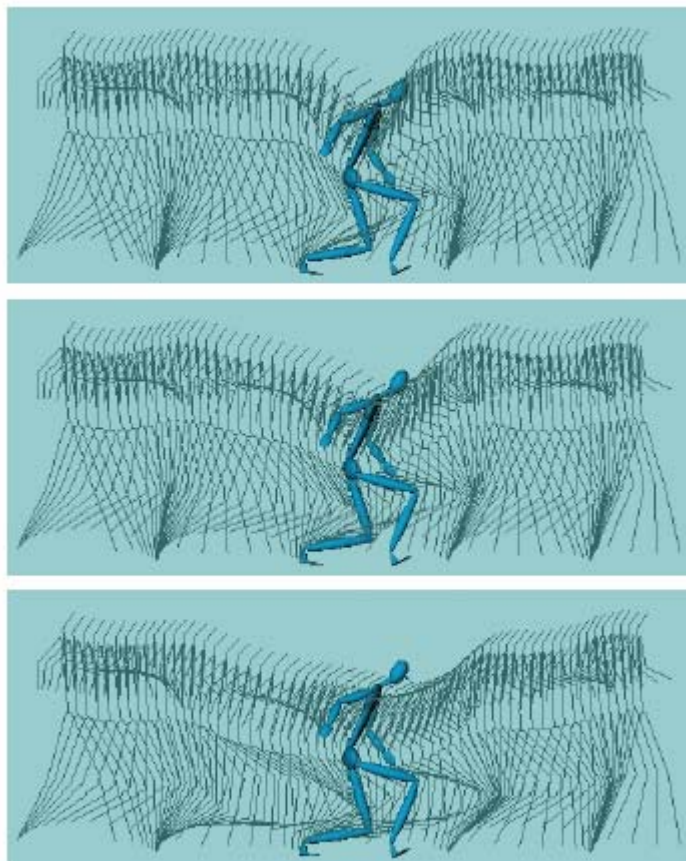
Apskaičiavimui  $d_k$ , kuris interpoliuoja perkėlimą, galima naudotis keliomis metodikomis. Šiuo atveju pasirenkamas kreivės derinimo metodas, nurodytas 1.15 formulėje. Šis pasirinktas metodas yra labai greitas, kadangi sprendimas gali būti pasiektas analitiškai priešingas nei kitas galimas metodo sprendimo skaitinis būdas.

Naudojant B-splaino kreivės derinimo metodą, iškyla problemos – rezultato kreivė gali būti mažiau tiksli ir gali truputį banguoti. Šis pasiūlytas algoritmas kompensuoja tokius trūkumus, kaip globalų skleidimo perkėlimą grubiame lygyje ir įvykdant vėlesnius suderinimus detaliam lygyje.

### 3.3.3. Mazgo išretinimas

Šis siūlomas metodo realizavimas dvigubina mazgo sekos tankumą nuo vieno lygio į sekanti. Jei  $\tau_k$  turi  $(n+3)$  valdančiųjų taškų, tuomet sekanti geresnė mazgo seka  $\tau_{k+1}$  turės  $(2n+3)$  valdančiųjų taškų. Mazgo sekos  $\tau_k$  tankumas,  $1 \leq k \leq h$ , apibrėžia apribojimo įtaką diapazonui perkėlimo žymėjimo lygyje  $k$ . Tai svarbu tiesioginiam valdymui. Vartotojo įvedimas yra interpretuojamas kaip apribojimų įvedimas, kurie yra nedelsiant pridedami į nurodytų apribojimų aibę. Tuomet, sistema glaudžiai deformuoja judėjimo klipų dalį aplink šį modifikuotą kadra. Čia įtakos diapazonas judėjimo klipse yra daugiausiai priklausantis  $\tau_k$  išretinimui. Didesnis išretinimas tarp mazgų duoda platesnį deformacijos diapazoną. 7 pav. gyvai užfiksuotas ėjimo judėjimas buvo interaktyviai modifikuojamas viduriniame kadre taip, kad personažas palinksta pirmyn ir nuleidžia dubenį. Žmogus yra pavaizduojamas modifikuotam kadre. Deformavimosi diapazonas yra apibrėžiamas mazgo sekų tankumu. Mazgas  $\tau_1$  yra retinamas kiekviename 4 kadre (viršutinis vaizdas), 6 (vidurinis vaizdas) ir 12 kadru (apatinis vaizdas) kiekiais. Be to, perkėlimo žymėjimas  $d_1$ , kuris yra išvestas iš grubios  $\tau_1$  sekos, turi nenulinę reikšmę dėl platesnio diapazono, vienisai skleidžiant

judėjimo pokytį. Sekantis smulkus perkėlimo žymėjimas  $d_k, 2 \leq k \leq h$ , atlieka sėkmingus suderinimus, kad atitiktų reikalavimus.



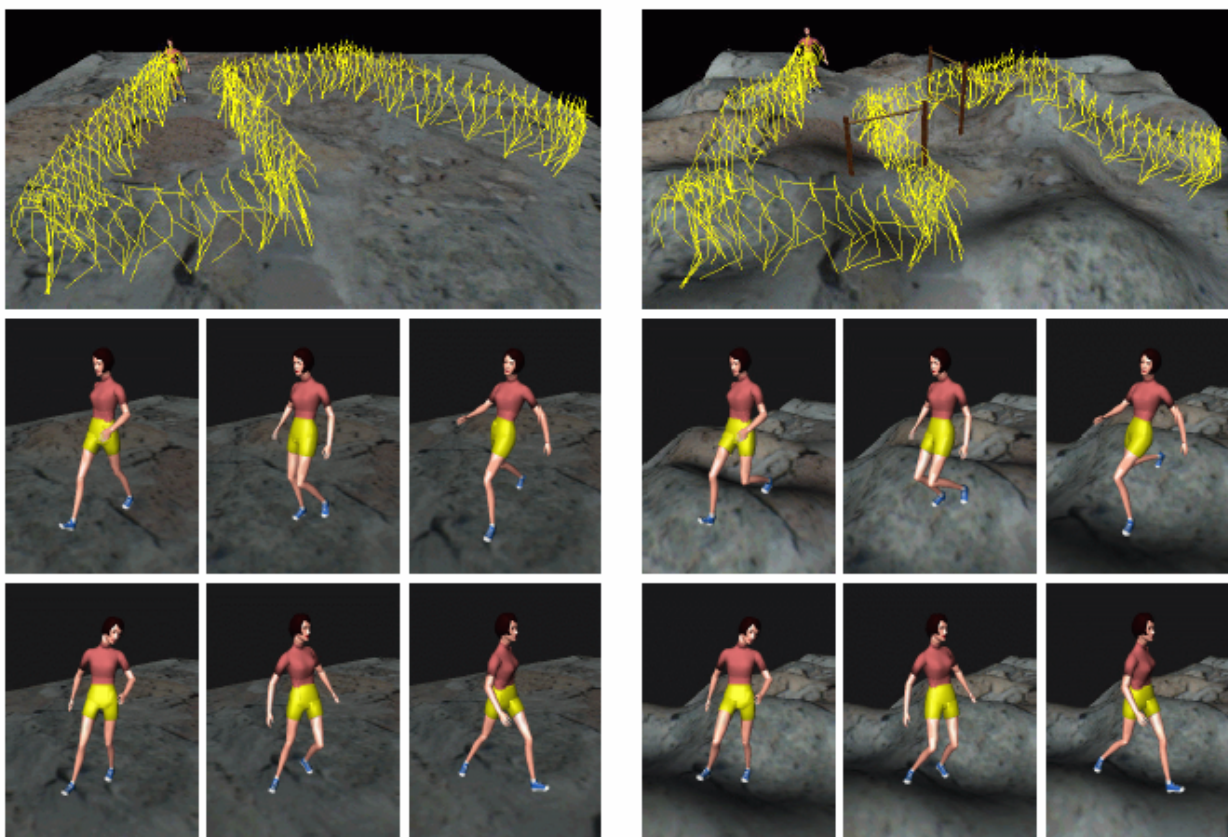
**7 pav.** Mazgų išretinimas, gaunant platesnį deformacijos diapazoną  
(Lee, 2001)

Smulkaus mazgo sekos  $\tau_h$  tankumas valdo baigtinio judėjimo  $m_h$  tikslumą. Jeigu  $\tau_h$  yra pakankamai smulkus sutalpinti apribojimų išsidėstymą laike,  $m_h$  gali tiksliai atitikti visus apribojimus. Šiame metode pakanka tik 5 lygių, kad gaunamas rezultatas būtų vizualiai tikroviškas.

### 3.3.4. Pradiniai spėjimai

Nagrinėjant įvykių erdvės problemą, norint pagerinti tiek skaitinį optimizavimo konvergavimą, tiek rezultato kokybę, geras pradinis spėjimas yra labai svarbus reikalaujamam sprendiniui. Šiame metode pradinis spėjimas judėjimo pritaikymui yra pradinio segmento padėties perkėlimas į pradinį judėjimą. Tai yra motyvuojama,

atsižvelgiant į vaikščiojimo judėjimą, kuris yra pritaikytas nelygiai vietai, kaip parodyta 8 pav. (a) ir (b) dalyse.



. a) Pradinis ėjimo judėjimas lygiu paviršium;

b) Pritaikytas judėjimas grublėtam paviršiui

**8 pav.** Personazo judėjimas lygiu ir gruoblėtu paviršiumi (Sung Shin, 2002)

Pėdos stovėsenos padėtis, kuri liečia vietovės paviršių, yra patempta aukštyn mažo kalnagūbrio link vietovėje ir personažas nepageidaujamai priverčiamas pritūpti. Net jeigu atvirkštinės kinematikos sprendinys bando minimizuoti junginio kampų nukrypimus, tai negali sutrukdyti kelio baigtiniam sulinkimui. Sumažinti šiam trūkumui, pakeičiama pradinio segmento pozicija atitinkamai geometriniais pokyčiams. Pradinis segmentas perkeliamas kiekviename kadre vidurkiu kontaktinių taškų pasikeitimo atžvilgiu. Pradinio segmento padėties pokytis kadre gali būti vientisai platinamas į šalia esančius kaimyninius kadrus, naudojant B-splaino atitinkamą metodą.

### **3.4. Teorinės dalies išvados**

1. Buvo pavaizduotas vienas iš naujesnių požiūrių į žmogaus judėjimą, tai pritaikant egzistuojančiam personažui, kai yra reikalaujamos savybės, specifikuojamos apribojimų aibe.

2. Buvo panaudoti daugialygiai B-splainai, nes šių splainų polinominė forma yra universali ir ši forma tinka bet kurios eilės splainams, išlieka ta pati ir skiriasi tik sumuojamų narių skaičiumi. Taigi naudojant B-splainus, galima sudaryti vieną universalų skaičiavimo procedūrą.

3. Hierarchinė struktūra kompensuoja galinius lokalinio aproksimavimo metodo trūkumus, skleidžiant perkėlimo žymėjimą grubiame lygyje ir vėliau priderinant tai smulkiuose lygiuose.

## 4. HIBRIDINIS ALGORITMAS

Norint gauti natūralų 3D erdvėje virtualaus žmogaus judėjimą, reikia įtraukti sudėtingus reiškinius, kurie gali apimti mechaniką, biomechaniką, fiziką, įvairius specifikuojimo, optimizavimo, modeliavimo, generavimo ir kt. procesus.

Personažo judėjimo lygčių matematiniai modeliai ir algoritmų pritaikymas dažniausiai yra grindžiami hibridiniais modeliais, kuriuose sąveikauja dviejų metodų idėjos.

Kadangi žmogaus judėjimas yra smarkiai kintanti ir reikalaujanti daug žinių bei galimybių sritis, tai norint gauti realistišką žmogaus judėjimo animaciją, siūlau įvertinti žmogaus judėjimą teoriniame lygmenyje.

Siūlomas algoritmas yra trumpai apibendrinamas: pirmiausia įvedami žmogaus pradinius judėjimo duomenis, kuriuose nurodomi apribojimai, fizikinės savybės, kaip pradinė padėtis, žmogaus masė, kūno dalių masės pasiskirstymas, judėjimo greitis bei pagreitis ir t.t. Pasinaudojus žmogaus masės svoriniu grafu, įvertiname judėjimo galimybių savybes. Toliau pasinaudojus šiais duomenimis, sukonstruojami tarpiniai judėjimo vaizdai, kurių apjungimas yra apskaičiuojamas, panaudojant daugialygius B-splainais. Pasinaudojus šiuo būdu, surandamos judėjimo sekos su trokštamomis, savybėmis, tenkinant nurodytus apribojimus. Po transformavimo veiksmų, šios sekos yra sujungiamos į vieną ir jeigu, kad gauti suplanuotą judėjimą.

Siūlomas hibridinis metodas yra aprašomas detaliau.

### 4.1. Apribojimų nustatymas

Svarbiausias žmogaus judėjimo proceso perkėlimo apribojimą galima įvardyti pėdos konstrukciją. Norint išsaugoti efektyvumą laike, personažo judėjime yra įtraukiamas paprastas metodas automatiškai aptinkamai kojos konstrukcijai:

1. Abi pėdos yra sužymimos kaip neapribotos kiekviename kadre;
2. Skaičiuojama dviejų charakteringų taškų vertikali padėtis  $p_y$  ir greitis  $|v|$  (vienas ant kulno, kitas ant kojos piršto) kiekvienai pėdai kiekviename kadre. Visi judėjimo duomenys yra įrašomi, kai žmogus juda lygiu pagrindu. Jei  $p_y$  ir  $|v|$  yra žemiau atitinkamų rėžių  $T_{p_y}$  (~10cm) ir  $T_v$  (~1,25cm/kadre) nurodytam F pėdos P taškui i-tajame kadre, tuomet pažymėsime dydį F kaip perkeltą į tašką P i-ajame kadre.

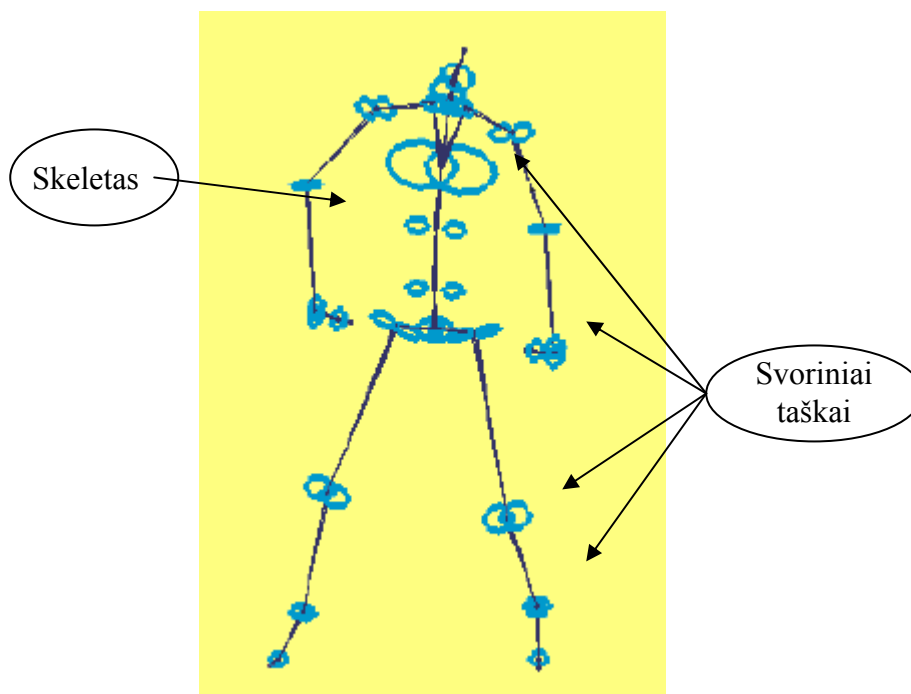
3. Apskaičiuojame  $a_{mc}$  – vertikalių kūno masės centro (mc) pagreitėjimą tiems kadrams, kurie vis dar yra nepriskirti (nėra pastatyta arti jokia pėda). Teoriškai jei  $a_{mc}$  yra didesnė nei gravitacijos konstanta (manoma, kad  $g$  neigiama), turėtų būti keletas aukštesnių jėgų kūno viduje, kurios reikš priskirtų taškų egzistavimą. Praktiškai galima būtų pasirinkti  $0.33g$  ( $0.33g$  pasirinkimas užtikrina, kad šis procesas neduos klaidų, nors gali būti išimčių), kaip suspaudimo reikšmę ir tuo pačiu galima dar kartą pakartoti 2 žingsnį su naujais režiais ( $T_{p_y}$  (~10cm) ir  $T_v$  (~2,5cm/kadre)) tuose kadruose, kuriuose, kurių  $a_{mc}$  yra didesnė nei  $0.33g$ .

4. Priskirsime visus neapribotus kadrus kaip “jokie”, kurie reikš, kad jokia pėda nėra pastatyta. Taigi kiekvienas kadras turės apribojimą.

5. Norint išlaikyti tam tikrą nenutrūkstumą, reikėtų pašalinti netikėtus apribojimo pokyčius. Tokius apribojimus, kurie tęsiasi trumpu laiko periodu, tarkim mažesniu nei  $F_c$  kadrai (dabar nustatyta 2), pakeičiame pirmą pusę kadru su jų pirmtakų apribojimais ir antra pusę jų tęsiniais. Šis procesas yra iteruojamas iki tol, kol nereikalingi jokie procesai.

Aukščiau paminėtame algoritme, baigtinis skirtumas yra naudojamas greičio/pagreičio skaičiavimams. Norint apskaičiuoti kūno masės centro padėtį, siūlau pridėti tam tikrų rekomendacinių taškų aibę (vadinsime “taškų rinkinį”), kad išskirti personažo modelio skeleto dalis. Šie taškai yra atitinkamai įvertinami svoriais ir paskirstomi visame kūne (9 pav.). Panaudojamas taškų rinkinys, kad efektyviai aproksimuoti personažo kūno masės paskirstymą skirtingose judėjimo pozose. Tokiu būdu, kūno masės centro padėtis yra paprastai įvertinama svoriniu erdviniu šių taškų vidurkiu.





9 pav. Svorinių taškų aibė personažo griaučiuose.  
Kiekvieno apskritimo centras yra taško padėtis ir kiekvieno apskritimo dydis pažymi to taško svorį

#### 4.2. Panašių reikšmių apskaičiavimas

Tam, kad apibrėžti, kiek yra panašūs du judėjimo klipai, skaičiuojama lygiuojami skirtumai tarp dviejų judėjimo pozų. Čia siūlau panaudoti „taškų rinkinio“ modelį. Per bet koki judėjimą „taškų rinkinys“ perkels kartu ir prie kūno pritvirtintas kūno dalis, kurių aproksimavimas duoda kūnui svorio pasiskirstymą skirtingose padėtyse. Todėl apibrėžiama panašumo funkcija  $D$ , kuri pažymi svorinio atstumo sumą tarp išrikiuotų kadro segmentų dviejų judėjimų klipų taško rinkinių:

$$D(A, i, B, j) = \sum_{t=0}^{L_f-1} \sum_{k=0}^{N_p-1} w_k \times D_p(A, i+t, B, j+t, k) \quad (3.1)$$

Kur

1.  $A$  ir  $B$  yra judėjimo klipai;
2.  $i$  ir  $j$  yra kadro numeriai;
3.  $L_f$  yra iš anksto apibrėžta konstanta, nurodanti kadro segmento ilgį, naudojamą panašiams skaičiavimams;
4.  $D_p(A, i, B, j, k)$  yra atstumas tarp dviejų „ $k$ -tųjų taškų“ atitinkamai  $A(i)$  ir  $B(j)$  kadru;
5.  $N_p$  yra taškų skaičius iš anksto apibrėžtame „taškų rinkinyje“;
6.  $w_k$  yra iš anksto apibrėžtas svoris, siejamas su „ $k$ -tuoju tašku“ „taškų rinkinyje“;

7.  $(A, i, B, j, L_f)$  turi būti suderinami. Kitu atveju  $D(A, i, B, j) = \text{inf}$ .

Apibrėžus panašumo funkciją, galima pastebėti, kad mažesnės reikšmės pažymi du judėjimus, kurie yra vienas į kitą panašūs. Kaip lygus perėjimas iš vieno į kitą reikalauja daugiau informacijos negu gali būti išgaunama atskirame kadre, taip naudojame  $L_f$  besitęsiančių kadro segmentą tam, kad efektyviai įtraukti svarbią kūno dalių kinematinę informaciją (greitis, pagreitis ir t.t.) į visumą. Norint pašalinti globalius transformacijos skirtumus (apsisukimas aplink vertikalią ašį ir įgyvendinimas (perskaičiavimas)), išrikiuojame dviejų „taškų rinkinių“ sekas, pagrįstas:

- 1) kūno masės centro padėtimis;
- 2) pirminio junginio pirmųjų dviejų kadro esamomis kryptimis. Šis išrikiavimo

procesas bus panaudojamas vėliau tiek perėjimo konstravime ir tiek judėjimo sintezėse.

**1 lentelė**

*Apribojimų suderinamos poros. K – kairė koja; D – dešinė; A- abi; n – nei viena*

Apribojimų poros	(K,K)	(D,D)	(A,A)	(N,N)	(K,A) arba (A,K)	(D,A) arba (A,D)	Kiti atvejai
Suderinamumas	Taip	Taip	Taip	Taip	Taip	Taip	Ne
Naujas apribojimas	K	D	Abi	Nei viena	Kairė	Dešinė	Nėra

3.1 formulėje, reikalaujama, kad  $(A, i, B, j, L_f)$  būtų galima suderinti. Tai reiškia, kad kiekviena atitinkamų segmentų kadro pora  $A(i), A(i + 1), \dots, A(i + L_f - 1)$  ir  $B(j), B(j + 1), \dots, B(j + L_f - 1)$  privalo turėti atitinkamus apribojimus, kurie apibrėžiami 1 lentelėje. Šis reikalavimas išvengia apribojimo dviprasmybės perėjimo sekose, kai norima sugeneruoti ir taip pagerinti perėjimo tęstinumą.

### 4.3. Pereinamų taškų parinkimas

Naudojant panašumo (suderinamumo) metriką aprašytą aukščiau, galima suskaičiuoti panašumo matricą dviem judėjimo A ir B klipams su kiekvienu perėjimu  $M_{ij}$  kaip  $D(A, i, B, j)$  reikšmė. Intuityviai šioje matricoje, žemesnės reikšmės duotas žingsnis pažymi aukštą kokybės perėjimo tašką ir viduje minimaliai reiškia geriausią perėjimo tašką iš visų kaimynų. Tačiau, skirtingos judėjimo rūšys turi skirtingas kinematinės savybes ir skirtingus perėjimo

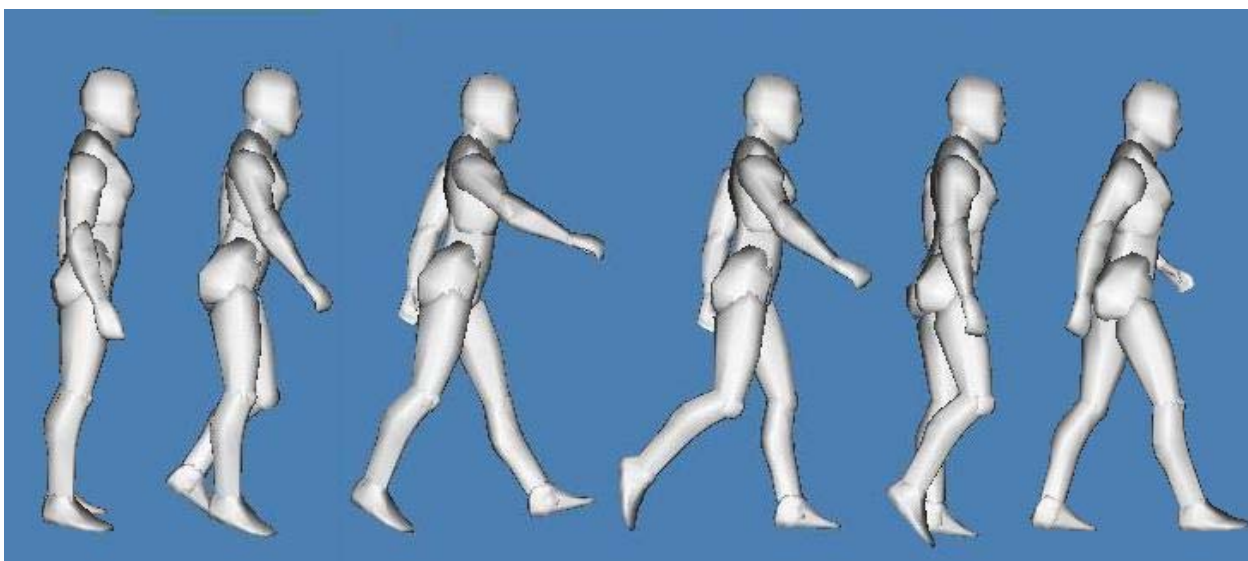
tikslumo reikalavimus, be kita ko reikia skirtingų žingsnių reikšmių. Tikiuosi, kad geriausias būdas rasti žingsnį yra pažiūrėti atgal į pradinis judėjimo klipus. Palyginant du kaimyninius kadro segmentus (tarkim 1~10 ir 2~11) tame pačiame judėjime, galima gauti būdingo panašumo kiekybinę sąvoką tarp skirtingų šio judėjimo padėčių, kurias pavadinsiu “panašumo charakteristikos” reikšmė  $R$ . Kadangi ši reikšmė gali keistis judėjimo klipo metu, todėl įvedamas palyginimas į žemesnio kadro lygį. Šis palyginimas gali būti apskaičiuotas, prijungiant tame pačiame judėjime A ir B klipams 3.1 formulėje. Taigi  $R$  gali būti aprašytas kaip

$$R(A, i) = D(A, i, A, i + F_{post}) \quad (3.2)$$

Kur  $F_{post}$  (poslinkis) yra apibrėžiamas iš anksto kaip konstanta, reiškiantis subalansuotus kadrus tarp segmentų panašumo charakteristikos reikšmės skaičiavimų. Dabar kiekvienas judėjimo kadras (išskyrus tuos, kurie yra judėjimo klipo pabaigoje) yra priskiriamas panašumo charakteristikos reikšmei, taigi galima palyginti pradinio judėjimo panašumo reikšmę  $D(A, i, B, j)$  su dvejomis atitinkamomis charakteristikų reikšmėmis:

$$D_{pal}(A, i, B, j) = \frac{D(A, i, B, j)}{\min(R(A, i), R(B, j))} \quad (3.3)$$

Kur  $\min(x, y)$  yra minimali funkcija. Šio palyginimo  $D_{pal}$  rezultatas yra tolygus matavimas, kuris parodo, koku panašumu dviejų kadro segmentai yra kartu su savo pradinio judėjimu. Dabar konvertuojamas kiekviena  $D$  matrica į naują  $D_{pal}$  matricą, ir parenkama lokaliai minimali reikšmė mažesnė nei konstantos žingsnis  $T_q$ . Šis žingsnis  $T_q$  yra vartotojo apibrėžta konstanta, kuri valdo visą perėjimo klipų kokybę.



10 pav. Žmogau judėjimo vaizdo atrinkimas dviejų kaimyninių kadru parinkimui

Priede yra išvardijamos bendrinės konstantos, kurios yra paminėtos tyrimo dalyje, ir jų reikšmės, kurios yra parinktos, atsižvelgiant į teorines bei praktines rekomendacijas bei tyrimus, pagrįsti mūsų eksperimentui.

#### 4.4. Perėjimo konstravimas interpoliuojant

Kiekvienam perėjimo taškui, rastam 4.3 skyriuje, judėjimas bus konstruojamas derinimo proceso metu.

Tarkime  $D(A, i, B, j)$  yra panašumo reikšmė judėjimo taškui. Pirmiausia, kadro segmentams  $A(i) \sim A(i + L_f - 1)$  ir  $B(j) \sim B(j + L_f - 1)$  reikia būti transformuotiems taip, kad pirmas kadras būtų suvienodintas, kad:

1. kūno masės centras būtų toje pačioje pozicijoje;
2. pirminė junginio orientacija (tik aplink vertikalią ašį) turėtų būti tokia pati.

Tuomet yra siūloma tiesiškai interpoliuoti (arba aproksimuoti pritaikant daugialygius B-splainus 4.5 skyrius) tarp dviejų suvienodintų kadro segmentų, kad apskaičiuotų kiekvieną naujo  $C$  perėjimo kadra  $p$  ( $0 \leq p < L_f$ ):

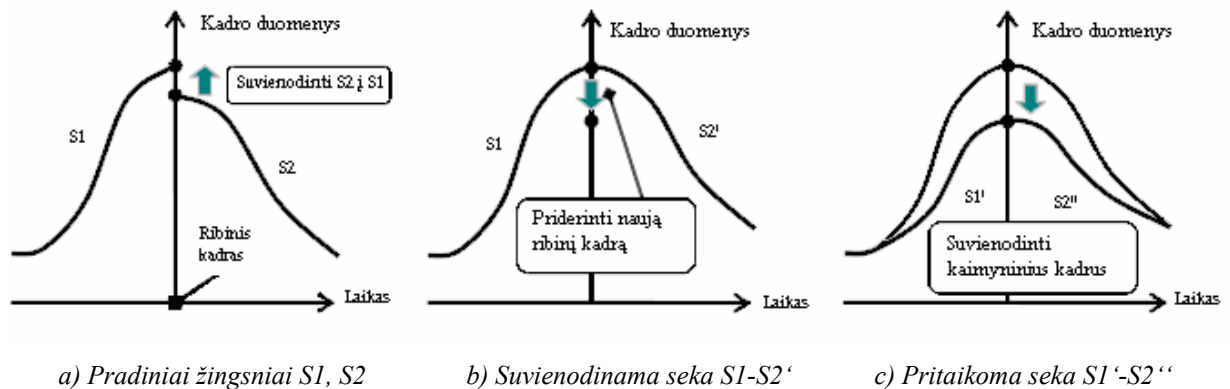
$$C(p) = \alpha\left(\frac{p}{L_f}\right) \times A(i+p) + \left(1 - \alpha\left(\frac{p}{L_f}\right)\right) \times B(j+p) \quad (3.4)$$

Kur  $\alpha(x)$  yra derinama svorio funkcija, apibrėžiama kaip:

$$\alpha = \begin{cases} 1, & x < 0, \\ 2x^3 - 3x^2 + 1, & 0 \leq x \leq 1 \\ 0, & x > 1 \end{cases} \quad (3.5)$$

Jei ši svorinė funkcija patenkina sąlygas  $\alpha(0) = 1, \alpha(1) = 0$ , ir  $\alpha'(0) = 1, \alpha'(1) = 0$ , tai judėjimo vientisumas gali būti išlaikomas abiejuose ribojimuose perėjimo judėjimo metu. Apribojimo tęstinumas yra visiškai garantuojamas nuo tų segmentų porų su nesuderinamais apribojimais, kurios jau buvo atmetos panašių reikšmių skaičiavimuose. 1 lentelėje buvo išvardytos naujo apribojimo pasirinkimas suderinamų porų naujiems perėjimo kadrams

Taigi suformavus nesusijusio judėjimo kadrus, toliau bandoma padaryti lankstų, vienodą ir lygų perėjimą iš vieno judesio į kitą, tam analizuojant žingsnio kitimo procesą (11 pav.).



11 pav. Ribinių kadrių suvienodinimas ir pritaikymas

Žingsniai visuomet yra apribojami kadrais, kurie prasideda nauju pėdos patalpinimo ant pagrindo apribojimu. Taigi tuo remiantis, galima atrinkti ribinius kadrus ir bandyti įtraukti kaimyninius kadrus tam, kad perėjimas nuo vieno kadro į kitą būtų kuo realistiškesnis, atitinkamai nenutrūkstantis. Neužbaigti žingsniai gali pasirodyti judėjimo sekos pabaigoje ir būti atmetami atmetimo procese.

Atmetimo procesas judėjimo procese yra atliekamas, išnagrinėjus kadre esančių mazgų perėjimus į kaimyninius taškus, atskiriant svarbiausius žingsnio mazgus. Jei šalia pereinamo mazgo nėra kaimyninio ir jis nėra įvertintas kaip svarbus, tai šį mazgą galima atmeti.

Toliau yra analizuojamas kiekvienas žingsnis ir saugoma mazgo judėjimo informacija, tokia kaip judėjimo greitis, krypties pasikeitimas, žingsnio gylis ir t.t.

#### **4.5. Perėjimo konstravimas aproksimuojant B-splainais**

Pasinaudosime Lee [28] pasiūlytu daugialygio B-splaino aproksimavimo būdu, norint surinkti kiekvieną perėjimo tašką į splaino paviršių. Atitinkamai konvertuosime duomenis į tam tikrą taškų judėjimo bei mazgų rinkinį ir panaudosime 3.2 skyriuje panaudotą algoritmą. Šiame algoritme įvesdami papildomą kūno masės centro įvertinamą taškų rinkinį, įvertinsime atitinkamai žmogaus judesio konstravimą.

#### **4.6. Galutinio judėjimo generavimas**

Suformavus naujo judėjimo seką, algoritmas gali sugrįžti atgal į pirmą žingsnį ieškodamas tikslesnio sprendinio su naujomis pradinėmis sąlygomis, jei gaunamą vaizdą norima pakoreguoti.

#### **4.7. Realizavimas**

Ši siūlomą algoritmą galima būtų realizuoti trimačių vaizdų kūrimo bei animavimo galimybes turinčiose programose, kaip Maya, pridėdant algoritmą kaip papildymą (plug-in), naudojant bet kurią nors programavimo kalbą (tarkim C++). Šis algoritmas siūlomas hibridinis algoritmas gali nuskaityti judėjimo sekų rinkinį, analizuoti jį ir generuoti realiaame laike besitęsiantį judėjimą su trokštamomis savybėmis.

Šis siūlomas algoritmas gali būti panaudotas kaip geresnės judėjimo sintezės alternatyva, vaizduojant daugiau lankstesnį, efektyvesnį, animuotą žmogų. Tačiau šis algoritmas nėra optimalus.

## IŠVADOS

1. Išnaginėjus žmogaus personažo judėjimo algoritmus bei metodus kompiuterinės animacijos srityje, galima pastebėti, kad metodai toliau vis sudėtingėja, reikalauja didesnių sąnaudų.
2. Labai žemo lygio abstrakcijoje animatorius gali būti labai tikslus, turėdamas nurodymus, kas turi būti parodyta ir kada. Labai aukštame abstrakcijos lygyje, animatorius gali tiesiog nurodyti “pasakyt kompiuteriui sukurti filmą apie šunį”. Šis lygis atlaisvina animatorių nuo žinojimo svarbos apie daugumą detalių, reikalingų, norint gauti animacijos fragmentą. Praktikoje animatoriai nori turėti galimybę dirbti skirtinguose lygiuose, priklausomai nuo norimo tikslo. Todėl išdirbis efektingų efektų įrankių gavimui, kurie leidžia animatoriams dirbti tiek aukštame, tiek žemame lygiuose abstrakcijų, yra labai svarbus.
3. Animatorius norėdamas vaizduoti judantį personažą, turi atsižvelgti ne vien tik į asmenines personažo savybes, įvairius apribojimus, bet taip pat atkreipti dėmesį į aplinką, jos pateikiamas kliūtis.
4. Pagrindinė problema šioje kompiuterinės animacijos srityje yra, kaip ir panaudojant kuo mažesnius išteklius, sudėlioti raktinius kadrus, kad žmogaus judėjimas būtų vaizduojamas ne tik tikroviškai, bet tuo pačiu sunaudojamos sąnaudos būtų kuo mažesnės.
5. Kompiuterio panaudojimas apskaičiuoti kiekvienam kadru aprašymui tarp raktinių kadru ir vaizduojamų atitinkamų kadru atvaizdu yra išspręsta, dažniausiai panaudojant B-splainus.
6. Buvo išnagrinėti du visiškai skirtingi metodai, priklausantys skirtingiems animuojamų personažų grupėms. Pirmajame vaizduojamas judėjimas yra grindžiamas fizikiniais dėsniais, dar vadinamais įvykių erdvėmis. Antrasis metodas, pagrįstas hierarchiniu B-splainų aproksimavimu.
7. Pristatomas hibridinis metodas, grindžiamas apribojimų įvedimu, kurių vienas iš pagrindinių yra žmogaus kūno dalių masės pasiskirstymas, konstruojami tarpiniai vaizdai, kurių apjungimas yra apskaičiuojamas arba tiesiškai interpoliuojant arba aproksimuojant daugialygiais B-splainais.

8. Algoritme stengiamasi pateikti tokį algoritmą, kuris padėtų vaizduoti kuo tikslesnį, efektyvesnį ir realistiškesnį žmogaus judėjimą 3D vaizde.
9. Siūlomas hibridinio algoritmo sprendimas nėra optimalus, tačiau bandoma išvengti sudėtingų skaičiavimų.



## LITERATŪRA

1. Plukas, K. Skaitiniai metodai ir algoritmai, 2001, 272 - 294 p. ISBN 9955-03-061-5.
2. Alexander, R. M.. Optimum take-off techniques for high and long jumps. 1990, p. 3-10.
3. Bodenheimer B.; Rose C. and etc. The Process of Motion Capture: Dealing with the Data. *Eurographics 1997: tarptautinės konferencijos pranešimų medžiaga. 1997* [žiūrėta 2005-12]. Prieiga per Internetą:  
<http://www.vuse.vanderbilt.edu/~bobbyb/pubs/dealingdata97.html>
4. Burtnyk, N.; Wein, M. Interactive skeleton techniques for enhancing motion dynamics in key frame animation . *Comumunication of ACM: tarptautinės konferencijos pranešimų medžiaga.. 1976*, p. 564 – 569. ISSN: 0001-0782.
5. Cohen, M. F. Interactive spacetime control for animation. *Computer Graphics / SIGGRAPH 1992: tarptautinės konferencijos pranešimų medžiaga. 1992*, 26 tomas: p. 293–302.
6. Cooley, P. The Essence of Computer Graphics. *Essence of computing*, 2001, p. 194 – 190.
7. De Leva, P. Adjustments to Zatsiorsky-Seluyanov’s segment inertia parameters. *Journal of Biomechanics: Biomechanikos žurnalas. 1996*, p. 1223–1230.
8. Foley, J. D.; van Dam. A.; and etc. Computer graphics principles and practice. 2nd edition in C. 1996, p. 1057 – 1064.
9. Gill, P. E.; Saunders, M.A.; Murray, W. SNOPT: An SQP algorithm for largescale constrained optimization. Technical Report. 1996.
10. Gleicher, M. Motion Path Editing. *ACM SIGGRAPH 2001: tarptautinės konferencijos medžiaga. 2001* [žiūrėta 2005-10]. Prieiga per Internetą:  
<http://www.siggraph.org/publications/newsletter/v33n4/contributions/gleicher.html>
11. Gleicher, M. Retargetting motion to new characters. *Computer Graphics / SIGGRAPH 1998: tarptautinės konferencijos pranešimų medžiaga. 1998*, 32: p. 33–42.

12. Kiss, S. Computer Animation for Articulated 3D Characters. 2002. [žiūrėta 2003-10]. Prieiga per Internetą: <http://www.cse.ohio-state.edu/~parent/animation/index1.html>
13. Lathrop, O. The way computer graphics works. 1997, 151 – 155 p.
14. Lee, J.; Shin, S. Y. A Hierarchical Approach to Interactive Motion Editing for Human-like Figures. *Computer Graphics / SIGGRAPH 1999: tarptautinės konferencijos pranešimų medžiaga*. 1999 [žiūrėta 2004-09]. Prieiga per Internetą: <http://mrl.snu.ac.kr/Papers/sig99.pdf>
15. Lee, S.; Chwa, K. Y.; Shin, S. Y. ir Wolberg, G.. Image metamorphosis using snakes and free-form deformations. *Computer Graphics / SIGGRAPH 1995: tarptautinės konferencijos pranešimų medžiaga*. 1995, 29: p. 439–448.
16. Mataric, M. J.; Zordan, V.; Mason, Z. Movement Control methods for Complex, Dynamically Simulated Agents: Adonis Dances Macarena. 1998 [žiūrėta 2003-12]. Prieiga per Internetą: <http://www.cs.berkeley.edu/~daf/games/webpage/motionpapers/aa98-wvictor.pdf>
17. Pearsall, D. J.; Reid, J.G.; Ross, R. Inertial properties of the human trunk of males determined from magnetic resonance imaging. *Annals of Biomed*. 1994, p. 692–706.
18. Pedotti, A.; Krishnan, V. V.; Stark, L. Optimization of muscle-force sequencing in human locomotion. 1978, p. 57–76.
19. Perent, P. Computer animation. Algorithms and techniques. 2004. p. 82 – 355. ISBN 5-93378-095-2.
20. Popovic, Z.; Witkin, A. Physically Based Motion Transformation. *Computer Graphics / SIGGRAPH 1999: tarptautinės konferencijos pranešimų medžiaga*. 1999. [žiūrėta 2004-09]. Prieiga per Internetą: [www.cs.washington.edu/homes/zoran/sigg99/preprint.pdf](http://www.cs.washington.edu/homes/zoran/sigg99/preprint.pdf)
21. Rose, C. Guenter, B. Bodenheimer, B and Cohen, M. Efficient generation of motion transitions using spacetime constraints. *Computer Graphics / SIGGRAPH 1996: tarptautinės konferencijos pranešimų medžiaga*. 1996, p. 147–154.
22. S. Lee, G. Wolberg, and S. Y. Shin. Scattered data interpolation with multilevel b-splines. *IEEE Transactions on Visualization and Computer Graphics: tarptautinės konferencijos pranešimų medžiaga*. 1997, 3(3): p. 228–244.

23. Shoemake, K. Animating rotation with quaternion curves. *Computer Graphics / SIGGRAPH 1985: tarptautinės konferencijos pranešimų medžiaga*. 1985, 19: p. 245–254, 1985.
24. Stuart Ferguson, R. *Practical Algorithms for 3D Computer Graphics*, 2001, p. 181 - 240.
25. Thalmann, N. M.; Thalmann, D. *Computer Animation in Future Technologies*. 1996 [žiūrėta 2005-12]. Prieiga per Internetą: [http://vrlab.epfl.ch/Publications/pdf/Magnenat\\_Thalmann\\_Thalmann\\_Interactive\\_CA\\_96.pdf](http://vrlab.epfl.ch/Publications/pdf/Magnenat_Thalmann_Thalmann_Interactive_CA_96.pdf)
26. Thalmann, N. M.; Thalmann, D. *Computer Animation: a Key Issue for Time Visualization. Accademical press: mokslinė spauda*. 1994 [žiūrėta 2004-11]. Prieiga per Internetą: [http://vrlab.epfl.ch/Publications/pdf/Magnenat\\_Thalmann\\_Thalmann\\_Scientific\\_Visualisation\\_94.pdf](http://vrlab.epfl.ch/Publications/pdf/Magnenat_Thalmann_Thalmann_Scientific_Visualisation_94.pdf)
27. W. Murray P. E. Gill and M. A. Saunders. SNOPT: An SQP algorithm for large-scale constrained optimization. *SIAM J. Optim.*, 12:979–1006, 2002
28. Witkin, A.; Kass, M. Spacetime constraints. *Computer Graphics / SIGGRAPH 1988: tarptautinės konferencijos pranešimų medžiaga*. 1988, 22: p. 159–168.
29. Witkin, A.; Popović, Z. Motion warping. *Computer Graphics / SIGGRAPH 1995: tarptautinės konferencijos pranešimų medžiaga*. 1995, 29: p. 105–108.
30. Zicheng Liu, Steven J. Gortler, and Michael F. Cohen. Hierarchical spacetime control. *Computer Graphics / SIGGRAPH 1994: tarptautinės konferencijos pranešimų medžiaga*. 1994.

## SANTRUMPŲ IR TERMINŲ ŽODYNAS

- Atvikštinė kinematika – tai procesas, kurio metu surandami lankstaus, dalimis sujungto objekto parametrai tam, kad objektas atsodurtų reikiamoje pozijoje.
- B–splainas – splaino funkcija, kuri turi mažiausią palaikymą atžvilgiu duoto laipsnio, glotnumo ir srities padalijimo
- DOF – (angl. degree of freedom) laisvės laipsnių skaičius.
- Hibridinis modelis – personažo judėjimo lygčių matematiniai modeliai ir algoritmų pritaikymas, kuriuose sąveikauja dviejų metodų idėjos.
- Interpoliacija – metodas, apibrėžiantis nurodytų taškų ar jų reikšmių skaičiavimus, kai taškai yra žinomi arba išdėstymą, naudojantis aplinkiniais taškais arba jų reikšmėmis.
- Kvaternionai – nekomutatyvių kompleksinių skaičių išplėtimas, kurie gaunami pridendant  $i, j, k$  prie realių skaičių. Galima išreikšti  $ai+bj+ck+d$  forma, kur  $a, b, c$  – realūs skaičiai.
- Splainas (poliliniija) – tai tolydžioji iki  $p$ -tosios eilės išvestinės imtinai funkcija, sudaryta iš kurios nors funkcijos dalių.
- SNOPT – yra programinės įrangos paketas, skirtas spręsti plačiu mastu optimizavimo problemoms (tiesinėms ir netiesinėms programoms)
- SQP metodas – (angl. sequence quadratic programming method) tolydus optimizavimo metodas, kuris reikalauja visų apribojimų ir tikslo funkcijų pirmų išvestinių apskaičiavimo, norint išspręsti įvykių erdvės optimizavimo problemą.

## PRIEDAS

### Hibridiniame algoritme minimos konstantos ir jų reikšmės

2 lentelė

Bendrinės konstantos bei jų reikšmės

Konstantos pavadinimas	Tipas	Siūloma pradinė reikšmė	Aprašymas
$T_{py}$	žingsnis	10 cm	Maksimalus vertikalus atstumas (tarp pagrindo ir dedamos kojos)
$T_v$	žingsnis	1,25 cm/kadre	Maksimalus greitis dedamai kojai
$G_{pri}$	žingsnis	0,33g	Minimalus kūno masės centro pagreitėjimas, pažymintis palaikančių jėgų egzistavimą
$F_c$	žingsnis	2 kadrai	Minimalus laiko periodas, per kurį turėtų tęstis apribojimas
$L_f$	konstanta	10 kadru	Kadro segmento ilgis naudojamas panašumo skaičiavimams
$N_p$	konstanta	45 taškai	Taškų skaičius apibrėžtas taškų rinkiniu
$w_k$ ( $k=1, 2, \dots, N$ )	konstanta	įvairūs pagal k	Svoris, susietas su k-tuoju tašku „taškų rinkinyje“
$F_{post}$	konstanta	1 kadras	Kadru poslinkis tarp segmentų panašumo nurodymo reikšmės skaičiavimas