

UNIVERZA V LJUBLJANI
Fakulteta za računalništvo in informatiko

David Klemenc

**RAZVOJ POSTOPKA ZA PROSTORSKO REKONSTRUKCIJO
RAZCEPIŠČA SRČNIH ARTERIJ NA OSNOVI SLIK KORONARNEGA
ANGIOGRAMA**

DIPLOMSKO DELO VISOKOŠOLSKEGA STROKOVNEGA ŠTUDIJA

MENTOR: prof. dr. Franc Jager

Ljubljana, 2012

Rezultati diplomskega dela so intelektualna lastnina Fakultete za računalništvo in informatiko Univerze v Ljubljani. Za objavlanje ali izkoriščanje rezultatov diplomskega dela je potrebno pisno soglasje Fakultete za računalništvo in informatiko ter mentorja.



Št. naloge: 00145/2011

Datum: 02.09.2011

Univerza v Ljubljani, Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Kandidat: **DAVID KLEMENC**

Naslov: **RAZVOJ POSTOPKA ZA PROSTORSKO REKONSTRUKCIJO
RAZCEPIŠČA SRČNIH ARTERIJ NA OSNOVI SLIK KORONARNEGA
ANGIOGRAMA**

**DEVELOPMENT OF A PROCEDURE FOR SPATIAL
RECONSTRUCTION OF CORONARY ARTERY BIFURCATION ON THE
BASIS OF CORONARY ANGIOGRAM IMAGES**

Vrsta naloge: Diplomsko delo visokošolskega strokovnega študija prve stopnje

Tematika naloge:

Razvijte postopek za trodimenzionalno rekonstrukcijo razcepišča koronarnih arterij med postopkom koronarografije za izboljšano ocenjevanje kotov med arterijami. Postopek naj temelji na geometričnih transformacijah in naj na osnovi rentgenskih slik koronarnega angiograma dobljenih pod različnimi koti v prostoru (dvodimenzionalne projekcije) postopno in z uporabo korekcijskih vektorjev zaradi zamaknjenih slik (relativna odstopanja položaja srca v prostoru) sestavlja trodimenzionalni model danega razcepišča koronarnih žil. Razviti postopek testirajte ob simulaciji tipičnih napak, ki se pojavljajo med postopkom zajemanja slik.

Mentor:

prof. dr. Franc Jager

Dekan:

prof. dr. Nikolaj Zimic



IZJAVA O AVTORSTVU

diplomskega dela

Spodaj podpisani **David Klemenc**,

z vpisno številko **63070521**,

sem avtor diplomskega dela z naslovom:

RAZVOJ POSTOPKA ZA PROSTORSKO REKONSTRUKCIJO RAZCEPIŠČA SRČNIH ARTERIJ NA OSNOVI SLIK KORONARNEGA ANGIOGRAMA

S svojim podpisom zagotavljam, da:

- sem diplomsko delo izdelal samostojno pod mentorstvom
prof. dr. Franca Jagra
- so elektronska oblika diplomskega dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko diplomskega dela
- soglašam z javno objavo elektronske oblike diplomskega dela v zbirki »Dela FRI«.

V Ljubljani, dne 15.03.2012

Podpis avtorja:

ZAHVALA

“Per aspera ad astra!”

Prof. dr. Francu Jagru se zahvaljujem za vse potrpljenje, ki ga je imel z menoj. Za idejo, ki me je vzpodbudila, da se lotim reševanja problema opisanega v diplomski nalogi in za informacije ter materiale, ki sem jih potreboval, pa se zahvaljujem doc. dr. Matjažu Klemencu dr. med..

Hvaležen sem tudi družini in prijateljem za spodbudo in pomoč pri pisanju diplomske naloge.

KAZALO

POVZETEK	8
KLJUČNE BESEDE	8
ABSTRACT	9
KEY WORDS	9
UVOD	10
1 KORONOGRAFIJA SRCA	14
2 GEOMETRIJSKE OSNOVE TRIDIMENZIONALNE REKONSTRUKCIJE ARTERIJSKIH BIFURKACIJ	16
2.1 <i>ROTACIJA TOČKE OKOLI OSI V 3D PROSTORU</i>	16
2.2 <i>PREMICA SKOZI TOČKO</i>	17
3 PROJEKCIJE IN KOTI	19
4 REKONSTRUKCIJA RAZCEPIŠČA	28
4.1 <i>RAZSTAVITEV TRIDIMENZIONALNE ROGOVILE NA PROJEKCIJE IN SESTAVA ROGOVILE IZ PROJEKCIJ</i>	28
4.2 <i>SESTAVLJANJE PROJEKCIJ KORONARK V TRIDIMENZIONALNI MODEL</i>	31
4.3 <i>ALGORITEM ZA REKONSTRUKCIJO RAZCEPIŠČA</i>	35
5 PRIMER	41
6 DEMONSTRACIJSKA KODA	42
6.1 <i>IZRAČUN PROJEKCIJ IN KOTOV</i>	42
6.2 <i>IZRIS ROGOVILE IN NJENIH PROJEKCIJ</i>	45
6.3 <i>REKONSTRUKCIJA ENE STRANICE Z METODO SEŠTEVANJA VEKTORJEV</i>	46
6.4 <i>REKONSTRUKCIJA ENE STRANICE Z METODO SEKANJA RAVNIN</i>	47
6.5 <i>PRIMERJAVA METODE SEŠTEVANJA VEKTORJEV IN METODE SEKANJA RAVNIN</i>	48
6.6 <i>REKONSTRUKCIJA RAZCEPIŠČA IN IZRAČUN KOTOV</i>	49

6.7 REKONSTRUKCIJA RAZCEPIŠČA IZ SLIK	53
7 RAZPRAVA	54
8 LITERATURA	55

KAZALO SLIK

Slika 1: balonski kateter	10
Slika 2: od leve proti desni: Judkinsov kateter za levo koronarno arterijo, Judkinsov kateter za desno koronarno arterijo, "pigtail" kateter za slikanje votline levega prekata	11
Slika 3: rentgenska slika desne koronarne arterije z označeno zožitvijo	12
Slika 4: rentgenska slika razcepišča oblike T (leva koronarna arterija)	12
Slika 5: rentgenska slika razcepišča oblike Y (leva koronarna arterija)	13
Slika 6: prikaz premikanja detektorja rentgenskih žarkov	14
Slika 7: leva koronarna arterija slikana pod koti $\alpha = 41^\circ$ $\beta = -27^\circ$, tik po iztisu	15
Slika 8: leva koronarna arterija slikana pod koti $\alpha = 41^\circ$ $\beta = -27^\circ$, tik pred iztisom	15
Slika 9: prokaz kota med vektorjema p_1 in p_2 v prostoru in kota med projekcijami le-teh.	19
Slika 10: rotacija p_1 in p_2 za $\pi/6$ okoli y osi	20
Slika 11: rotacija p_1 in p_2 za $\pi/3$ okoli y osi	21
Slika 17: prikaz rogovile, z modro barvo je označen vektor SA, z rdečo vektor SB, zeleno pa je označen vektor SC	28
Slika 18: ortogonalne projekcije rogovile na ravnine $x = 0$, $y = 0$ in $z = 0$	28
Slika 19: rogovila, njene projekcije na ortogonalne ravnine ter presečišča premic, ki predstavljajo projekcije iste točke	29
Slika 20: zelena črta prikazuje rekonstrukcijo stranice SC s pomočjo seštevanja vektorjev, istoležna rdeča črtkana črta pa originalno SC stranico, kot vidimo je rekonstrukcija s seštevanjem vektorjev točna	30
Slika 21: prikaz razcepišča proksimalne koronarne arterije (MB proximal) v stransko vejo (SB) in distalno koronarno arterijo (MB distal) (kri teče od MB proximal proti MB distal in SB)	31

Slika 22: razcepišče koronarne žile slikano pod koti $\alpha = 47^\circ$ $\beta = 29^\circ$	32
Slika 24: rekonstrukcija stranice rogovile s presečiščem dveh ploskev, črtkana črna črta prikazuje rezultat.....	33
Slika 25: primerjava dveh metod rekonstrukcije rogovile, zelena črta predstavlja metodo seštevanja vektorjev (ta zaradi napačnega podatka o dolžini enega izmed projiciranih vektorjev odstopa), oranžna prikazuje rezultat pridobljen z metodo sekanja ravnin.....	35
Slika 26: rekonstrukcija rogovile z upoštevanjem napak zaradi lege detektorja rentgenskih žarkov (na vsaki projekciji so točke zamaknjene z vektorjem, ki simulira napako detektorja) in napak pri označevanju dolžine posameznega vektorja (eden izmed vektorjev na projekciji ima popačeno dolžino). Modre barve je originalna rogovila, ki predstavlja dejansko razcepišče žil, oranžne barve je rogovila, ki smo jo rekonstruirali iz projekcij.	40
Slika 27: rekonstrukcija razcepišča prikazanega na slikah 22 in 23, z modro je označena proksimalna glavna arterija (PM), z zeleno distalna glavna arterija (DM), z rdečo pa stranska veja (SB); kot med PM in DM je 167° , kot med DM in SB pa 47°	41

KAZALO TABEL

Tabela 1: koti projekcij	27
--------------------------------	----

POVZETEK

Bolezni srca in ožilja so najpogostejši vzrok smrti v razvitem svetu, saj zaradi njih umre približno dva milijona prebivalcev letno. Med boleznimi srca in ožilja predstavljajo najpomembnejši delež bolezni koronarnih arterij. Koronarne arterije oskrbujejo srce s kisikom in hranili.

Poseben problem koronarnih bolezni predstavljajo razcepišča koronarnih arterij, saj se zožitve največkrat pojavijo prav na teh mestih. Razcepišče je sestavljeno iz treh delov: proksimalne glavne arterije, stranske veje in distalne glavne arterije. Koti, ki jih te tri žile opisujejo, predvsem kot med glavno distalno arterijo in stransko vejo, vplivajo na potek zdravljenja:

- žičkanje stranske veje (uvajanje vodilnih žic v arterije) ter stentiranje lahko postane težavno, kadar je kot med distalno glavno arterijo in stransko vejo večji od 90°;
- premik karine pri stentiranju glavne arterije lahko povzroči zožitev začetnega dela stranske veje, kadar je kot manjši od 40°-50°.

Plitve kote med stransko vejo in glavno proksimalno arterijo lahko povežemo z večjo incidenco okluzije stranske veje, vendar obenem nam takšen kot omogoča lažje žičkanje stranske veje in lažjo postavitev stenta.

Tehnike trenutno predlagane za določitev kota med žilami, ki tvorijo razcepišče, delujejo tako, da se izmeri kot v projekciji z ocenjeno najmanjšo prikrajšavo žil. [1] S tridimenzionalno rekonstrukcijo razcepišča žil, ki je predmet diplomske naloge, lahko natančno določimo kote med žilami razcepišča in tako zmanjšamo napako, ki se zgodi zaradi popačitve kota v dvo-dimenzionalni projekciji.

V prvem poglavju opisujemo značilnosti koronarnega angiograma in težave, ki nastopijo zaradi utripanja srca. Drugo poglavje opisuje geometrijske osnove, ki jih bomo uporabljali pri tridimenzionalni rekonstrukciji razcepišča. Naslednje poglavje nam prikaže probleme s popačitvijo kota v različnih projekcijah. V četrtem poglavju bomo postopoma rekonstruirali razcepišče, začeli bomo s projekcijami brez napak in končali z rekonstrukcijo osnovano na projekcijah z več napakami. Peto in šesto poglavje predstavljata praktični del, v petem je prikazan primer rekonstrukcije razcepišča iz dveh slik, v šestem pa je koda, ki omogoča primerno natančno določitev kotov med arterijami.

Ugotovil sem, da lahko iz različnih dvodimenzionalnih projekcij istega razcepišča koronarnih arterij sestavim tridimenzionalen model razcepišča, kar omogoča večjo natančnost pri izračunu kota med žilami, ki razcepišče tvorijo.

KLJUČNE BESEDE

koronarografija, razcepiščni kot, tridimenzionalna rekonstrukcija

ABSTRACT

Cardiovascular diseases are the most important cause of morbidity and mortality in the developed countries. It's estimated that about two million people die because of heart disease. Coronary artery disease (CAD) presents the most important part of cardiovascular diseases. Coronary arteries are conduits responsible for nutrition and oxygenation of heart muscle cells.

Coronary bifurcations present special problem in CAD, because of high prevalence of stenoses - i.e. arteriosclerotic plaques. Bifurcation of coronary artery can be divided in three parts: proximal main artery (or main branch), distal main artery and side branch. The angles between branches are important as they dictate the way of dilatation and stenting:

- wiring of side branch can be difficult if the angle between distal main and side branch is greater than 90° , the same is true for stenting of ostial lesion of side branch;
- shift of carina can obstruct the ostial part of side branch after "provisional" stenting (i.e. stenting of proximal and distal part of main branch with covering of ostial part of side branch) if the angle is less than 40° - 50° .

Shallow angle between the proximal main vessel and side branch may be associated with higher incidence of side branch occlusion. However, shallow angle allows easier access for side branch wiring and stent delivery.

Techniques that are currently proposed for determining the angle between arteries, forming the bifurcation, consider the widest angle in the least foreshortened projection. [1] With 3-dimensional reconstruction of the bifurcation, which is the purpose of this work, we can accurately determine the angles between blood vessels forming the bifurcation, thus reducing the error that occurs due to subjective impression of the interventional cardiologist based on non orthogonal projection of the bifurcation.

In the first chapter we describe the characteristics of coronary angiogram and the problems that arise due to the heart movements. Geometrical formulas, which will be used for 3-dimensional reconstruction of bifurcations are presented in second chapter. Next chapter is dedicated to the analysis of the distortion of the original angle in various projections. In the fourth chapter, we will gradually reconstruct the bifurcation, we'll start with the reconstruction based on projections without error and finish with the reconstruction of the bifurcation based on projections with multiple errors. The fifth and sixth chapters show a practical example reconstructing a bifurcation from two projections and the code that solves the problems mentioned up to this point.

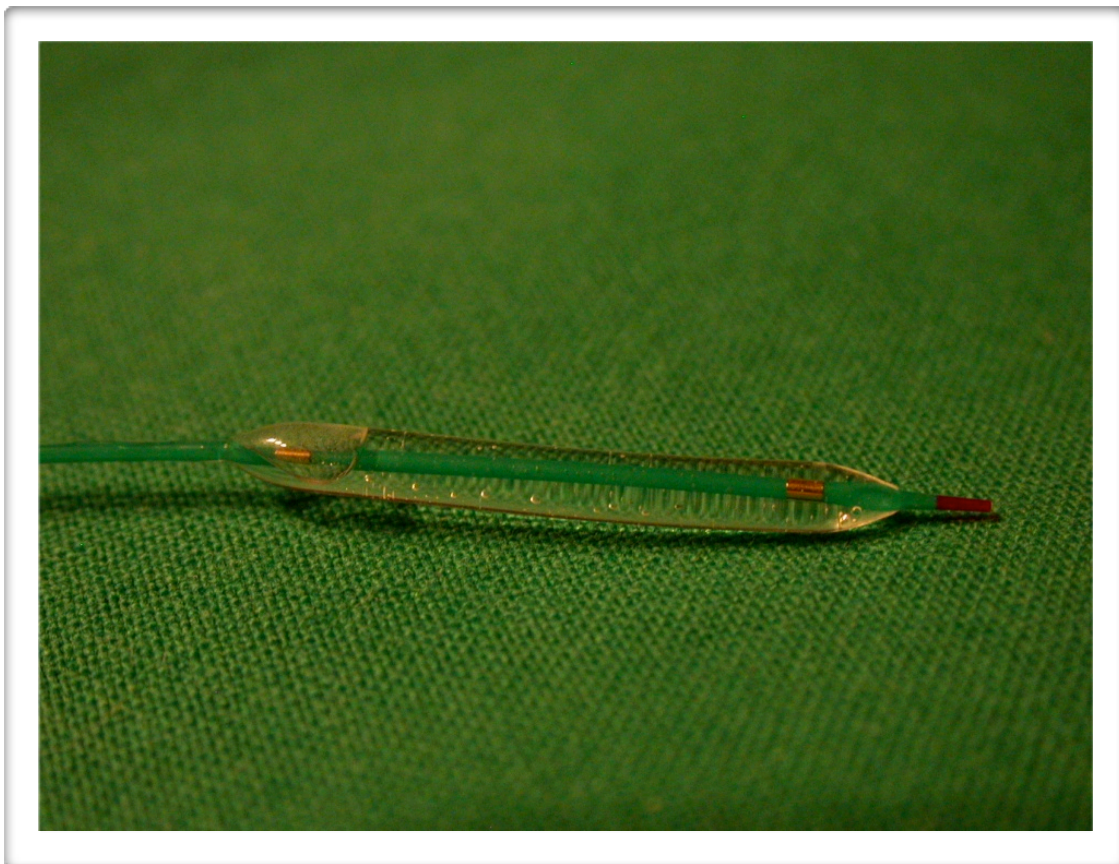
In conclusion, I found that it's possible to accurately construct a 3-dimensional model of coronary artery bifurcations by applying different standard 2-dimensional projections used during percutaneous coronary interventions.

KEY WORDS

coronarography, bifurcation angle, three-dimensional reconstruction

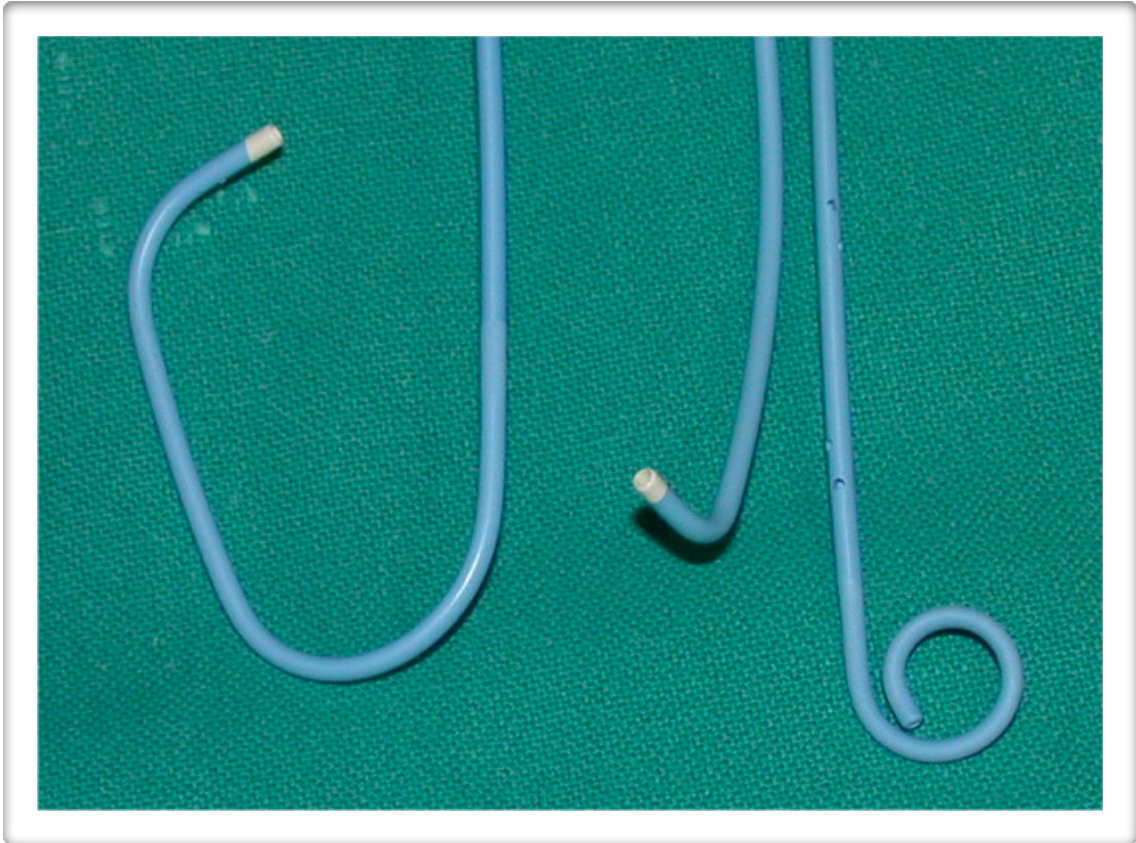
UVOD

Bolezni srca in ožilja so najpogostejši vzrok smrti v razvitem svetu. Strokovnjaki Svetovne zdravstvene organizacije (SZO) predvidevajo, da bodo tudi leta 2020 na prvem mestu v razvitem svetu, prav tako v razvijajočem se svetu. V zadnjih desetletjih se je pristop k zdravljenju precej spremenil, predvsem obravnava bolnikov s koronarno boleznijo. Mirovanje in preprosto antiagregacijsko zaščito z acetil salicilno kislino so najprej dopolnila zdravila, ki raztapljajo krvne strdke. Patomorfološke raziskave so pokazale, da je v večini primerov za zaporo koronarne arterije kriv prav krvni strdek. Opisana zdravila so bistveno izboljšala možnost preživetja bolnikov s koronarno boleznijo, še vedno pa je v približno 40% ostala koronarna arterija zaprta. V sedemdesetih letih prejšnjega stoletja je švicarski zdravnik Andreas Gruntzig začel zdraviti koronarno bolezen tako, da je v zoženo arterijo postavil poseben kateter, ki je imel na koncu majhen balonček (slika 1), s katerim je razširil zožitev.



Slika 1: balonski kateter

Danes je to najbolj učinkovit način zdravljenja različnih oblik koronarne bolezni, kot so stabilna angina pectoris, miokardni infarkt z ali brez elevacije ST spojnice. Vsako leto se po svetu opravi več kot dva milijona perkutanih koronarnih intervencij (PCI : percutaneous coronary intervention), v Sloveniji več kot 4000. Za dober izid posega je poleg izkušenosti operaterja in materialov pomemben tudi natančni prikaz koronarnih arterij. Koronarno arterijo prikažemo tako, da preko posebno oblikovanega katetra (npr. Judkinsov kateter za levo in desno koronarno arterijo prikazan na sliki 2) vbrizgamo kontrast v ustje koronarne arterije.

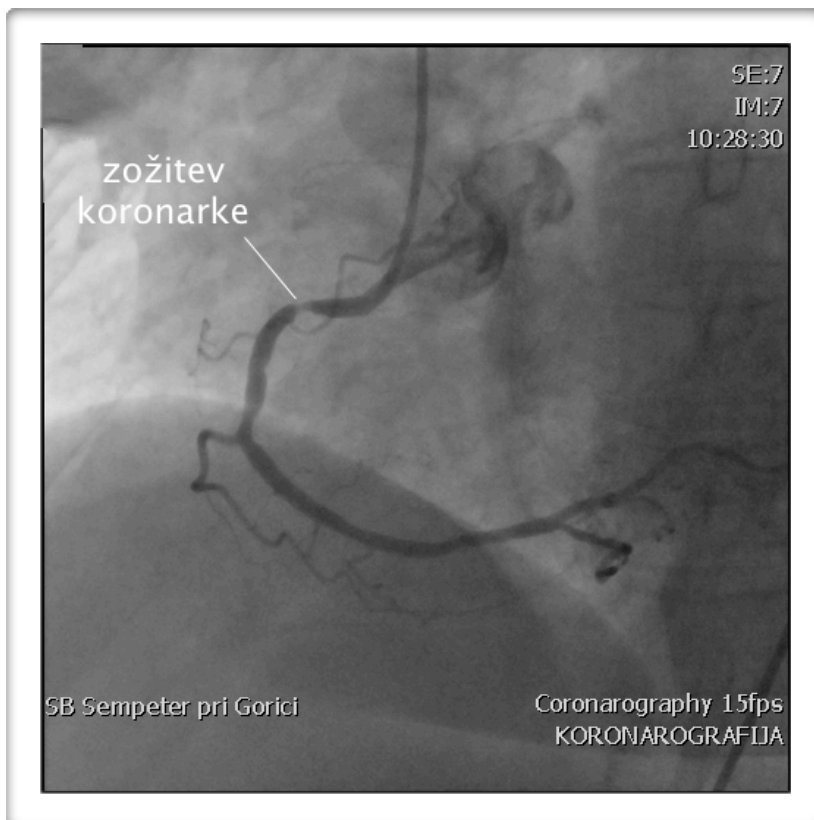


Slika 2: od leve proti desni: Judkinsov kateter za levo koronarno arterijo, Judkinsov kateter za desno koronarno arterijo, "pigtail" kateter za slikanje votline levega prekata

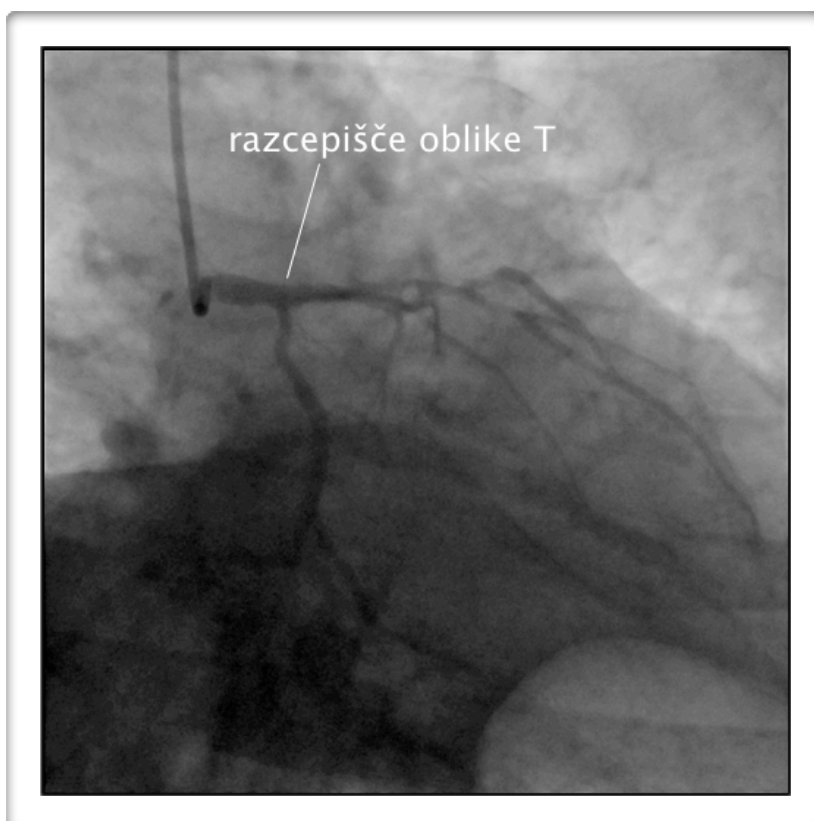
Z zmogljivim rentgenskim aparatom nato posnamemo krajšo sekvenco, 2-3 sekunde, s frekvenco 15 slik na sekundo. Posnetek, ki ga dobimo nam dejansko pokaže projekcijo odlitka svetline koronarne arterije na določeno površino. Na ta način lahko razmeroma natančno ocenimo mesto in stopnjo zožitve posamezne arterije (slika 3).

Med interventnimi posegi na koronarnih arterijah predstavljajo velik problem razcepišča. Približno 20% interventnih posegov pri zdravljenju koronarne bolezni je prav na mestih, kjer se večja arterija razcepi v dve manjši. Glede na kot med glavno žilo in stransko vejo lahko razcepišča (oz. bifurkacije) razdelimo na Y obliko - kot je $< 70^{\circ}$ (slika 5) in T obliko (slika 4), kjer je kot $\geq 70^{\circ}$. Analize posnetkov s pomočjo računalniške tomografije so pokazale, da je glavna bifurkacija na levi koronarni arteriji med levo sprednjo descendentno vejo in cirkumfleksno vejo največkrat T oblike, ostale pa so Y oblike.

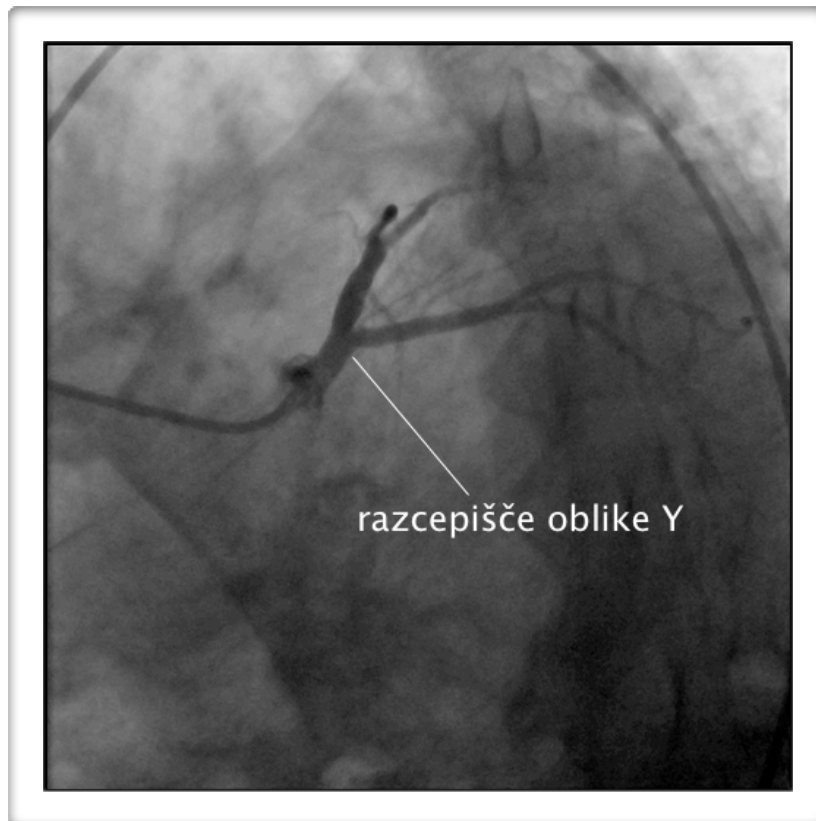
Študije, ki so do sedaj objavljale meritve kotov med žilami v razcepišču, niso imele standardiziranih metod meritev, zaradi česar so medsebojno neprimerljive. Lansky s sodelavci je predlagal metodologijo za ocenitev kota, ki temelji na merjenju kota med proksimalno glavno arterijo in stransko vejo, ter kota med distalno glavno arterijo in stransko vejo iz projekcij, v katerih so žile kar najmanj prikrajšane. Trenutno predlagane projekcije za optimalen prikaz razcepišča so osnovane na izkušnjah kardiologov, saj zaenkrat še ni znanstvenih študij iz tega področja. [1] Takšne meritve kotov se danes zelo pogosto uporabljajo in so široko sprejete.



Slika 3: rentgenska slika desne koronarne arterije z označeno zožitvijo



Slika 4: rentgenska slika razcepišča oblike T (leva koronarna arterija)

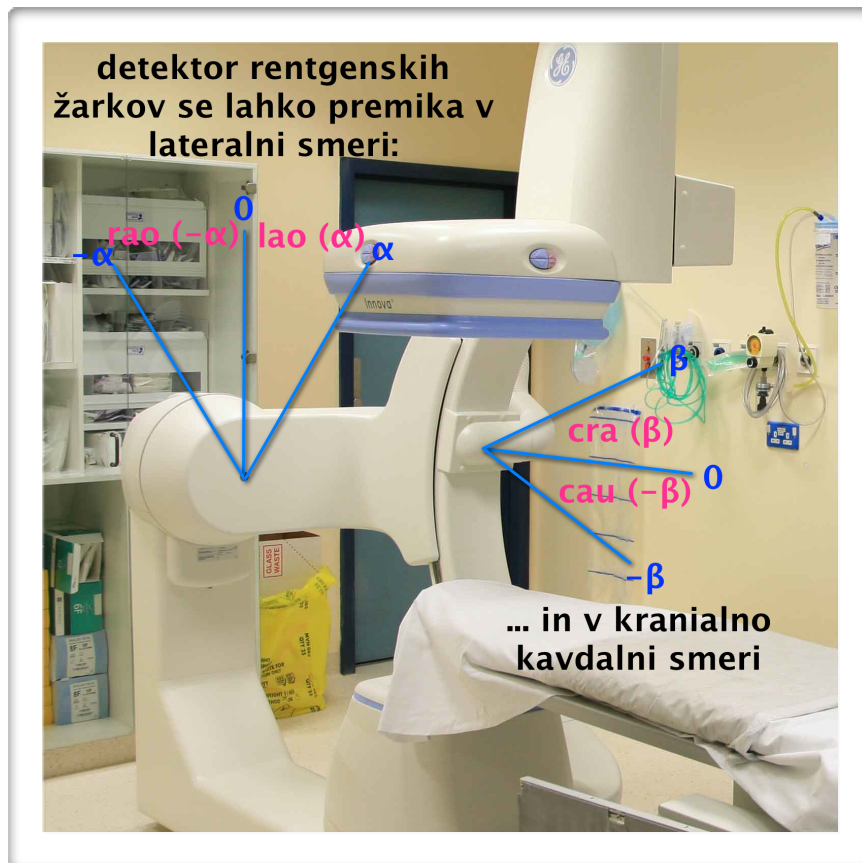


Slika 5: rentgenska slika razcepišča oblike Y (leva koronarna arterija)

1 KORONOGRFIJA SRCA

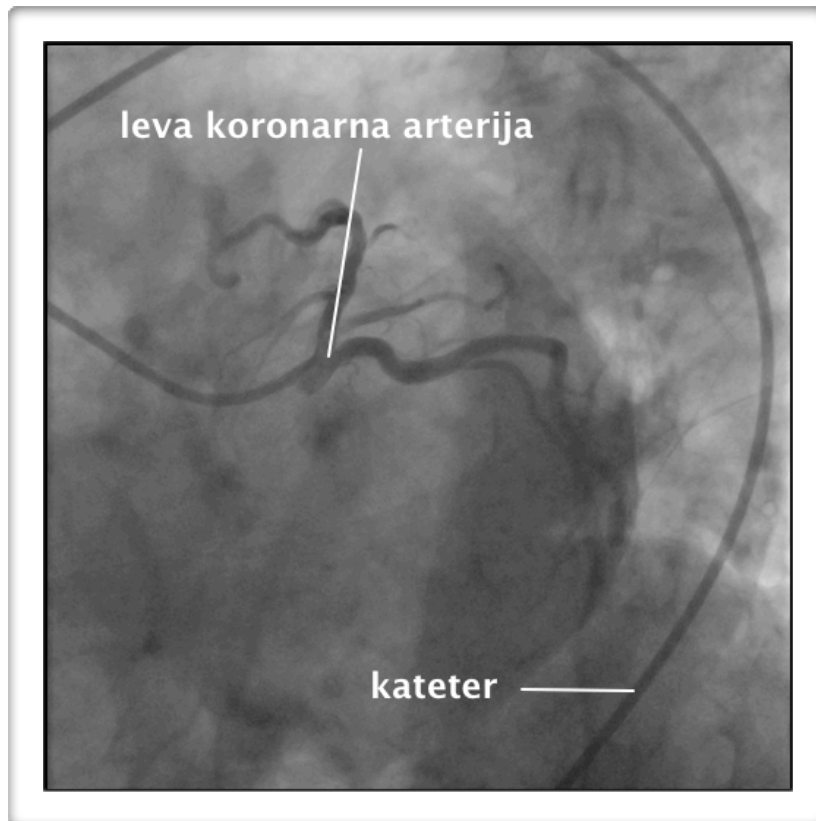
Koronarografija (slikanje srčnih žil) je preiskava, ki jo kardiologi vršijo, ko želijo dobiti natančen vpogled v anatomijo koronarnih arterij. Za to preiskavo se odločijo pri bolnikih, ki imajo ishemično bolezen srca in je zdravljenje z zdravili premalo učinkovito. Pri preiskavi ugotovijo ali bi bolniku lahko izboljšali zdravstveno stanje s perkutanim posegom ali z kiruršim posegom (aortokoronarnimi obvodi). Preiskavo vršijo pri budnem bolniku. Trajanje preiskave je različno, predvsem je odvisno od položaja ustja koronarnih arterij in kompleksnosti zožitev.

Koronarni angiogram je rentgenska slika, ki jo dobimo tako, da v srčne žile (koronarne arterije) vbrizgamo kontrastno sredstvo. Posnetki, pridobljeni na ta način, so dvodimenzionalna projekcija srčnih žil na ravnini določeni z dvema kotoma, ki nam povesta odklon detektorja v kranialno - kaudalni smeri ter lateralni smeri (slika 6). Pri tem pa pride do popačitve oziroma izgube informacij o dolžini žile in velikosti kota med žilami. Dolžina žil se v projekciji lahko samo zmanjša, medtem ko se kot lahko zmanjša do 0° ali poveča do 180° .

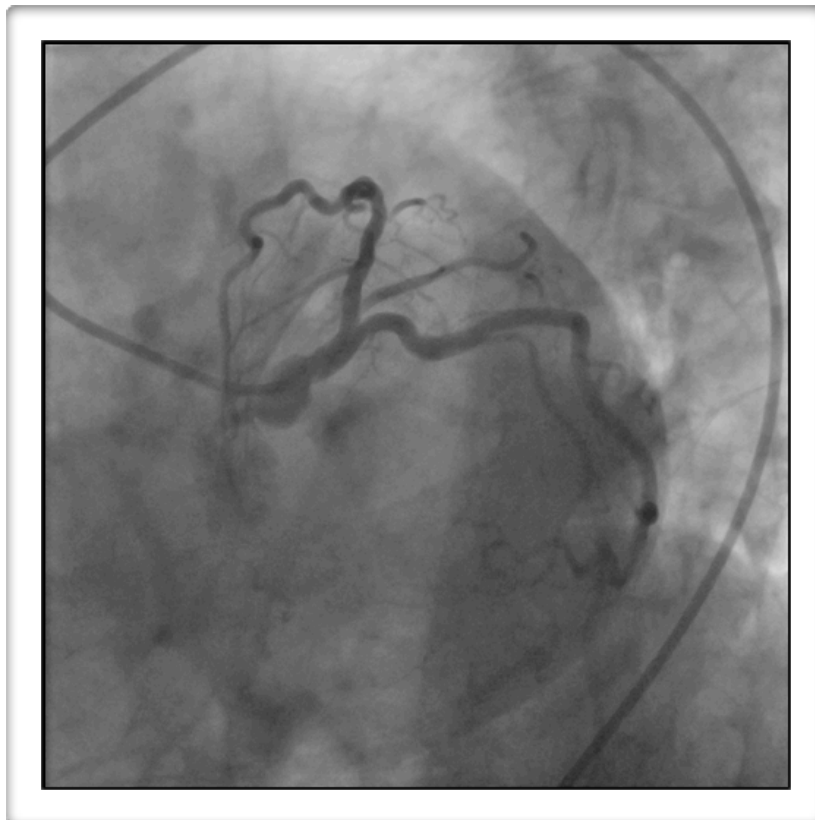


Slika 6: prikaz premikanja detektorja rentgenskih žarkov

Za vsako izbrano projekcijo napravimo 30 do 45 slik na katerih je vidnih več srčnih utripov. Ker se srce med snemanjem premika, se tudi dolžina žil in kot med njimi spreminja. Da bi kar najbolje ocenili dejanski kot med žilama je torej potrebno v različnih projekcijah izbrati sliko, ki prikazuje srce v enaki fazi srčnega ciklusa (npr.: ob koncu diastole, ki sovpada z R zobcem v EKG posnetku). Najlažje ločljiva položaja srca sta tik pred iztisom (konec diastole) (slika 8) in tik po iztisu (začetek diastole) (slika 7), ko je srce najmanj deformirano.



Slika 7: leva koronarna arterija slikana pod koti $\alpha = 41^\circ$ $\beta = -27^\circ$, tik po iztisu



Slika 8: leva koronarna arterija slikana pod koti $\alpha = 41^\circ$ $\beta = -27^\circ$, tik pred iztisu

2 GEOMETRIJSKE OSNOVE TRIDIMENZIONALNE REKONSTRUKCIJE ARTERIJSKIH BIFURKACIJ

Rentgenski posnetek koronarnih arterij nam daje dva podatka o ključnih točkah razcepišča, lego točke na sliki (x,y) in kotoma (položaj detektorja glede na preiskovanca) pod katerim je bila slika narejena (α, β) . S pomočjo danih podatkov je mogoče izračunati ravnino slike in točke na njej postaviti v tridimenzionalni prostor.

2.1 ROTACIJA TOČKE OKOLI OSI V 3D PROSTORU

Premik okoli osi Z:

$$x' = x \times \cos \theta - y \times \sin \theta$$

$$y' = x \times \sin \theta + y \times \cos \theta \quad (1)$$

$$z' = z$$

Premik okoli osi X:

$$y' = y \times \cos \beta - z \times \sin \beta$$

$$z' = y \times \sin \beta + z \times \cos \beta \quad (2)$$

$$x' = x$$

Premik okoli osi Y:

$$z' = z \times \cos \gamma - x \times \sin \gamma$$

$$x' = z \times \sin \gamma + x \times \cos \gamma \quad (3)$$

$$y' = y$$

Z zgoraj opisanimi enačbami lahko točke, ki jih dobimo kot koordinate slike (x,y) spremenimo v tridimenzionalne točke prostora s koordinato $z = 0$ $(x, y, 0)$ in nato z rotacijo preko x in y osi, ki ustreza položaju detektorja rentgenskih žarkov dobimo točke v pravilni ravnini.

Primer rotacije točke $(16, 16, 0)$ okoli osi x za 30° oziroma $\pi/6$ radianov:

Transformacijska funkcija napisana v obliki matrike:

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & \frac{\sqrt{3}}{2} & -\frac{1}{2} \\ 0 & \frac{1}{2} & \frac{\sqrt{3}}{2} \end{pmatrix}$$

Rezultat:

$$\left(16, 8\sqrt{3}, 8 \right)$$

2.2 PREMICA SKOZI TOČKO

V tridimenzionalnem prostoru ima premica, ki gre skozi točko (x_0, y_0, z_0) in je vzporedna z neničelnim vektorjem $v = (a, b, c)$ parametrično enačbo:

$$\begin{aligned}x &= x_0 + at \\y &= y_0 + bt \quad (4) \\z &= z_0 + ct\end{aligned}$$

zgoščeno lahko to napišemo (t je skalar):

$$x = x_0 + vt \quad (5)$$

(Neničelni smerni vektor lahko v naših primerih dobimo tako da točko $(0,0,1)$ zavrtimo preko osi x & y ali pa izračunamo normalo ravnine na kateri točka leži.)

Razdalja med dvema premicama v prostoru:

$$\begin{aligned}p_1 &= a + \mu b \\p_2 &= c + \nu d \quad (6) \\r &= (a - c) \cdot \frac{b \times d}{\|b \times d\|}\end{aligned}$$

V zgornjih enačbah so a in c točki, c ter d sta vektorja, μ in ν pa skalarja.

Ker bo večina premic zaradi napak prisotnih v podatkih mimobežnic nas bosta v nadaljevanju zanimali točki, ki sta si na mimobežnicah najbližji. Slednje dobimo tako da premico p_1 premaknemo v smeri vektorja, ki je ortogonalen na obe premici (le-ta opisuje ravnino, ki bi jo premici sestavljali, če bi se sekali), tako da se bo sekala s premico p_2 .

Vektor, ki nam da smer premika:

$$u = \frac{b \times d}{\|b \times d\|} \quad (7)$$

Dolžina premika, kot že izražena zgoraj:

$$r = (a - c) \cdot u \quad (8)$$

Premik premice:

$$p_1' = ru + a + \mu b \quad (9)$$

Sedaj lahko izenačimo premico p_1' in p_2 in dobimo iskane točke:

$$\begin{aligned} p_1' &= p_2 \\ ru + a + \mu b &= c + vd \end{aligned} \quad (10)$$

Za rešitev spremenljivk μ in v lahko uporabimo katerikoli dve komponenti vektorjev in naredimo dve linearni enačbi z dvema neznankama.

Seveda nas bo za natančno rekonstrukcijo razcepišča zanimala aritmetična sredina dobljenih točk. Aritmetično sredino bomo izračunali z enačbo 11:

$$\left(\frac{x_1 + x_2}{2}, \frac{y_1 + y_2}{2}, \frac{z_1 + z_2}{2} \right) \quad (11)$$

Formula za aritmetično sredino poljubnega števila tridimenzionalnih točk, nam bo pomagala, ko bomo z uporabo treh ali več projekcij poskusili izboljšati rezultate:

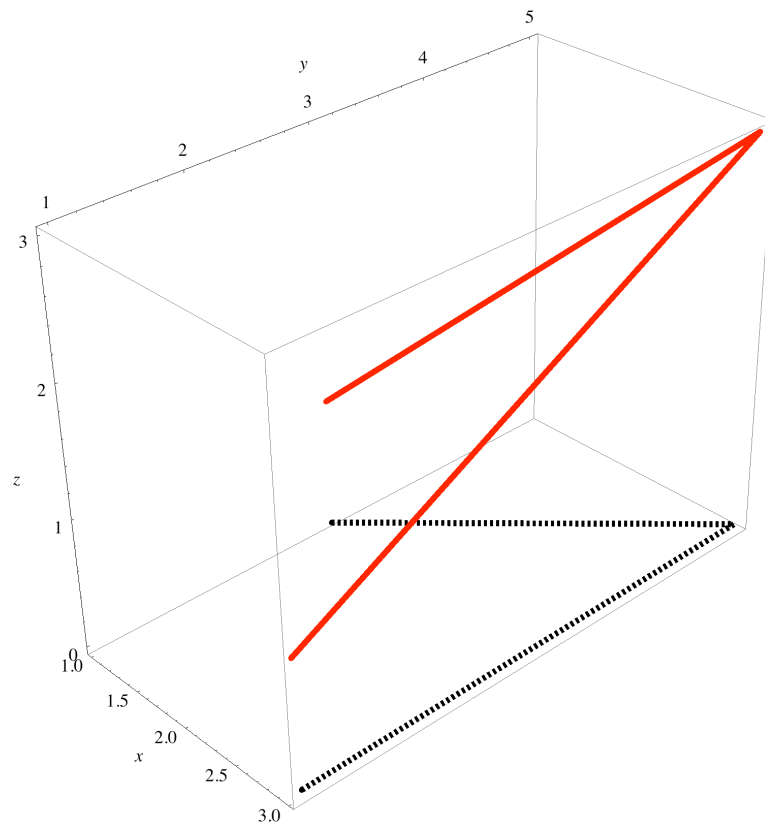
$$\left(\frac{\sum_{i=1}^n x_i}{n}, \frac{\sum_{i=1}^n y_i}{n}, \frac{\sum_{i=1}^n z_i}{n} \right) \quad (12)$$

S pravkar opisanimi enačbami lahko dobimo točko, ki je najboljša aproksimacija presečišču dveh (ali več) nesekajočih se premic, to bomo uporabili pri iskanju prave lege sredine razcepišča v prostoru.

3 PROJEKCIJE IN KOTI

Kot med dvema vektorjema v prostoru se le redko ujema (samo kadar sta vektorja vzporedna z ravnino na katero ju projiciramo) s kotom njune projekcije na poljubno določeno ravnino. Za primer vzemimo tri točke v prostoru $T_1(1,3,1)$, $T_2(3,5,3)$ in $T_3(3,1,1)$ projekcijo teh treh točk na ravnini $z = 0$ dobimo enostavno tako, da s komponento spremenimo v 0 torej $T_1'(1,3,0)$, $T_2'(3,5,0)$ in $T_3'(3,1,0)$. Tri točke nam lahko predstavljata dva vektorja v prostoru vektorja s smerjo $p_1(2,2,2)$ ter $p_2(0,4,2)$, projekciji vektorjev pa imata smeri $p_1'(2,2,0)$ ter $p_2'(0,4,0)$. Kot med vektorjema lahko izračunamo s formulo:

$$\cos(\theta) = \frac{p_1 \cdot p_2}{\|p_1\| \|p_2\|} \quad (13)$$



Slika 9: prokaz kota med vektorjema p_1 in p_2 v prostoru in kota med projekcijami le-teh

V zgornjem primeru (slika 9) znaša kot med vektorjema p_1 in p_2 0,684719 radianov ter kot med vektorjema p_1' in p_2' 0,785398 radianov. Prišlo je do popačitve kota, slednjo pa lahko z rotacijo prvotnih točk T_1 , T_2 in T_3 oziroma z rotacijo vektorjev p_1 in p_2 poljubno spremenimo. Ko prvotna vektorja p_1 in p_2 zavrtimo za $\pi/6$ radianov okoli y osi dobimo nova vektorja:

$$p_{1(y \rightarrow \pi/6)} = (1 + \sqrt{3}, 2, -1 + \sqrt{3})$$

$$p_{2(y \rightarrow \pi/6)} = (1, 4, \sqrt{3})$$

Novi smeri vektorjev smo dobili z rotacijsko matriko. Za vsako os vrtenja lahko enostavno izračunamo rotacijske matrike s spodnjimi formulami:

$$R_x(\alpha) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & \sin \alpha \\ 0 & -\sin \alpha & \cos \alpha \end{pmatrix}$$

$$R_y(\beta) = \begin{pmatrix} \cos \beta & 0 & -\sin \beta \\ 0 & 1 & 0 \\ \sin \beta & 0 & \cos \beta \end{pmatrix} \quad (14)$$

$$R_z(\gamma) = \begin{pmatrix} \cos \gamma & \sin \gamma & 0 \\ -\sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

V našem primeru dobimo rotacijsko matriko:

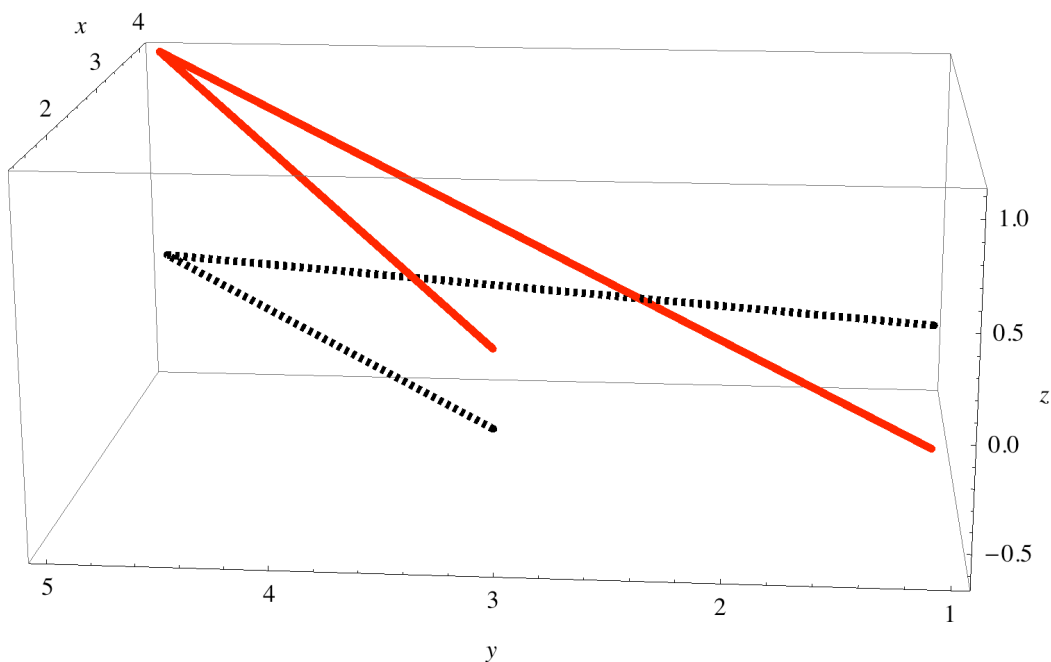
$$\begin{pmatrix} \frac{\sqrt{3}}{2} & 0 & \frac{1}{2} \\ 0 & 1 & 0 \\ -\frac{1}{2} & 0 & \frac{\sqrt{3}}{2} \end{pmatrix}$$

Projekciji $p1'_{(y \rightarrow \pi/6)}$ in $p2'_{(y \rightarrow \pi/6)}$ še vedno dobimo zgolj z zamenjavo z komponente za 0 torej:

$$p1'_{(y \rightarrow \pi/6)} = (1 + \sqrt{3}, 2, 0)$$

$$p2'_{(y \rightarrow \pi/6)} = (1, 4, 0)$$

Sedaj se celoten izgled (slika 10) in koti spremenijo:



Slika 10: rotacija $p1$ in $p2$ za $\pi/6$ okoli y osi

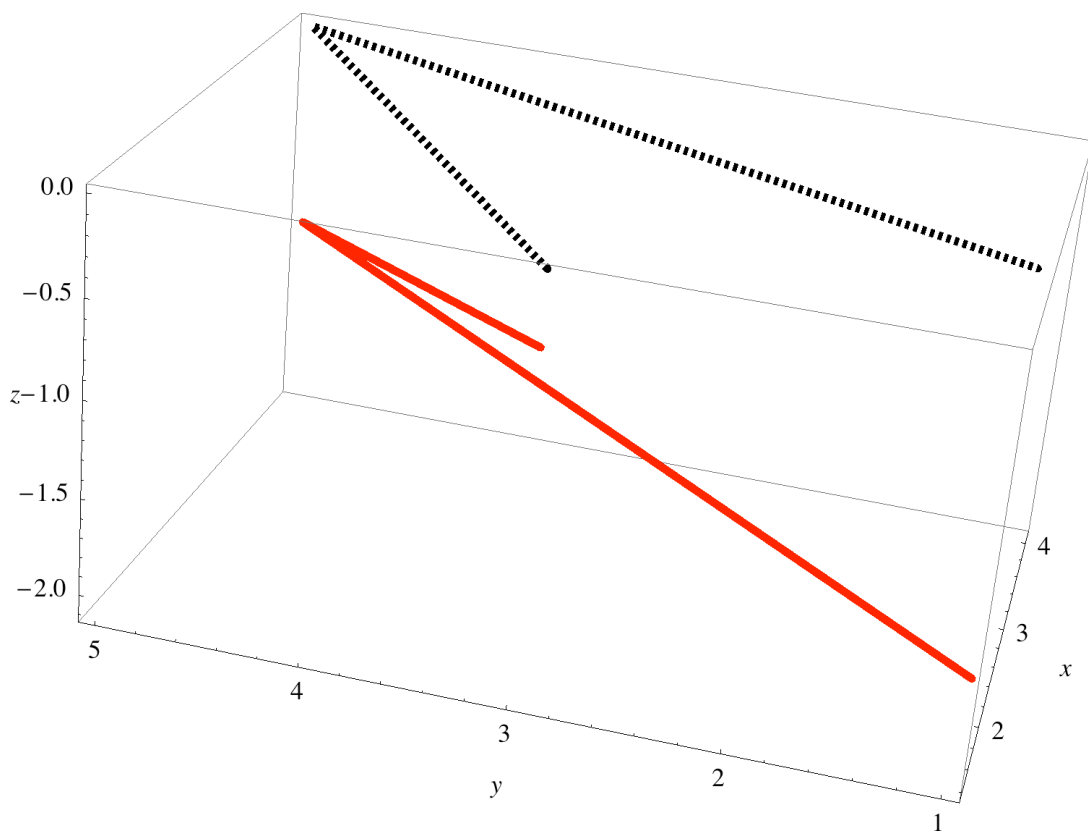
Kot med rotiranimi vektorjema je še vedno 0,684719 radianov, kot med projekcijama rotacij pa je sedaj 0,693903 radianov. Naslednjo rotacijo naredimo z rotacijsko matriko:

$$\begin{pmatrix} \frac{1}{2} & 0 & \frac{\sqrt{3}}{2} \\ 0 & 1 & 0 \\ -\frac{\sqrt{3}}{2} & 0 & \frac{1}{2} \end{pmatrix}$$

Ta nam prvotna vektorja $p1$ in $p2$ zavrti okoli y za $\pi/3$ radianov, tako da dobimo nova vektorja $p1_{y \rightarrow \pi/3}$ in $p2_{y \rightarrow \pi/3}$ prikazana rdeče v sliki 11:

$$p1_{(y \rightarrow \pi/3)} = (1 + \sqrt{3}, 2, 1 - \sqrt{3})$$

$$p2_{(y \rightarrow \pi/3)} = (3, 4, 1)$$



Slika 11: rotacija $p1$ in $p2$ za $\pi/3$ okoli y osi

Kot med njima seveda ostane nespremenjen 0,684719 radianov, projekciji na $z = 0$ ravnini pa vpenjata kot 0,530244 radianov.

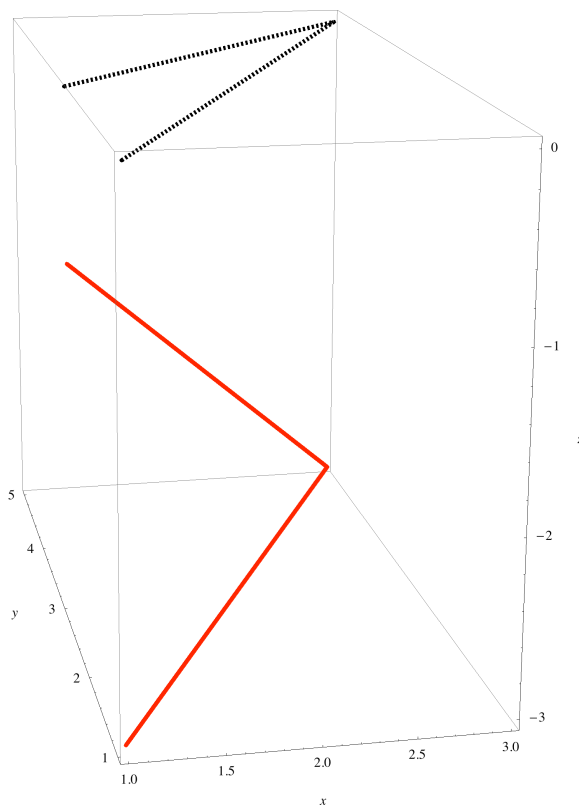
Nadaljno vrtenje okoli osi y za kot $\pi/2$ nam še izraziteje spremeni kot (slika 12). Rotacijska matrika je tokrat enostavno:

$$\begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{pmatrix}$$

Nova vektorja pa:

$$p1_{(y \rightarrow \pi/2)} = (2, 2, -2)$$

$$p2_{(y \rightarrow \pi/2)} = (2, 4, 0)$$



Slika 12: rotacija $p1$ in $p2$ za $\pi/2$ okoli y osi

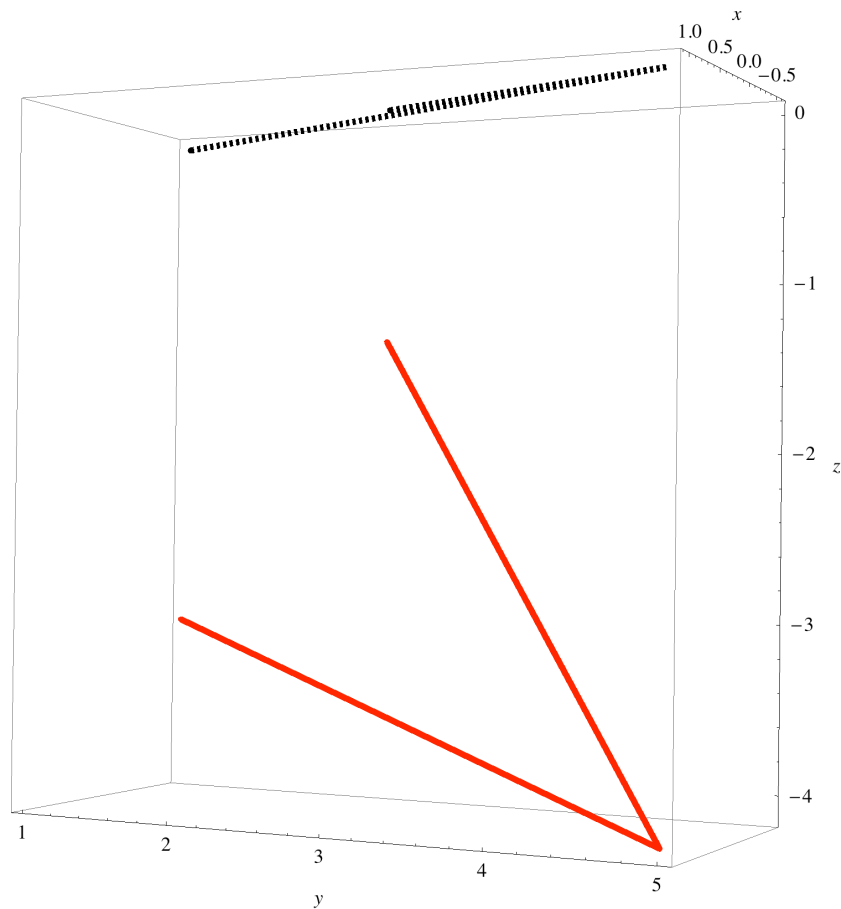
Projekciji imata kot vedno z komponento enako 0, kot med njima pa je tokrat 0,321751 radianov. Rotacija prvotnih vektorjev $p1$ in $p2$ za kot $2\pi/3$ okoli osi y nam prikaže kako lahko kot približamo 0 radianom oziroma en vektor popolnoma prekrijemo z drugim (slika 13). Rotacijska matrika ima sedaj naslednjo obliko:

$$\begin{pmatrix} -\frac{1}{2} & 0 & \frac{\sqrt{3}}{2} \\ 0 & 1 & 0 \\ -\frac{\sqrt{3}}{2} & 0 & -\frac{1}{2} \end{pmatrix}$$

Vektorja pa imata naslednji smeri:

$$p1_{(y \rightarrow 2\pi/3)} = (-1 + \sqrt{3}, 2, -1 - \sqrt{3})$$

$$p2_{(y \rightarrow 2\pi/3)} = (\sqrt{3}, 4, -1)$$



Slika 13: rotacija p1 in p2 za $2\pi/3$ okoli osi y

Kot med projekcijama je sedaj le 0,0577584 radianov.

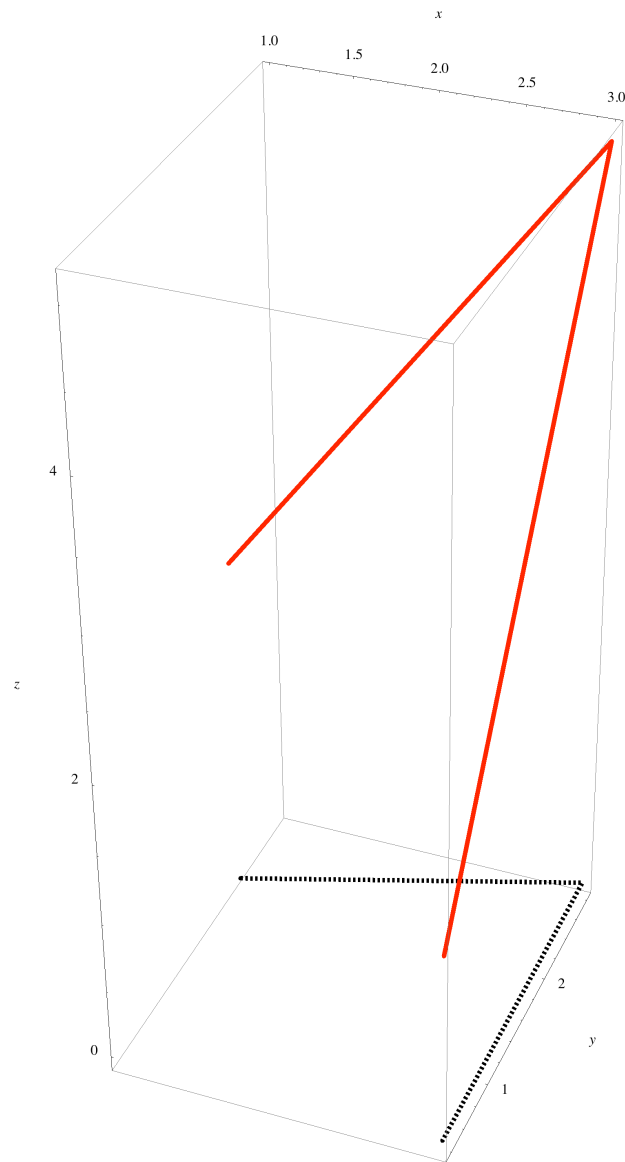
Rotacije okoli osi x v našem primeru predvsem povečujejo kot, ki ga opisujeta projekciji vektorjev p1 in p2 (povečanje in pomanjšanje kota je odvisno od prvotnih vektorjev, ne od osi rotacije). Rotacijo okoli osi x za kot $\pi/6$ radianov lahko tvorimo z matriko:

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & \frac{\sqrt{3}}{2} & -\frac{1}{2} \\ 0 & \frac{1}{2} & \frac{\sqrt{3}}{2} \end{pmatrix}$$

Z njo dobimo vektorja, prikazana na sliki 14:

$$p1_{(x \rightarrow \pi/6)} = (2, -1 + \sqrt{3}, 1 + \sqrt{3})$$

$$p2_{(x \rightarrow \pi/6)} = (0, -1 + 2\sqrt{3}, 2 + \sqrt{3})$$



Slika 14: rotacija $p1$ in $p2$ za $\pi/6$ okoli osi x

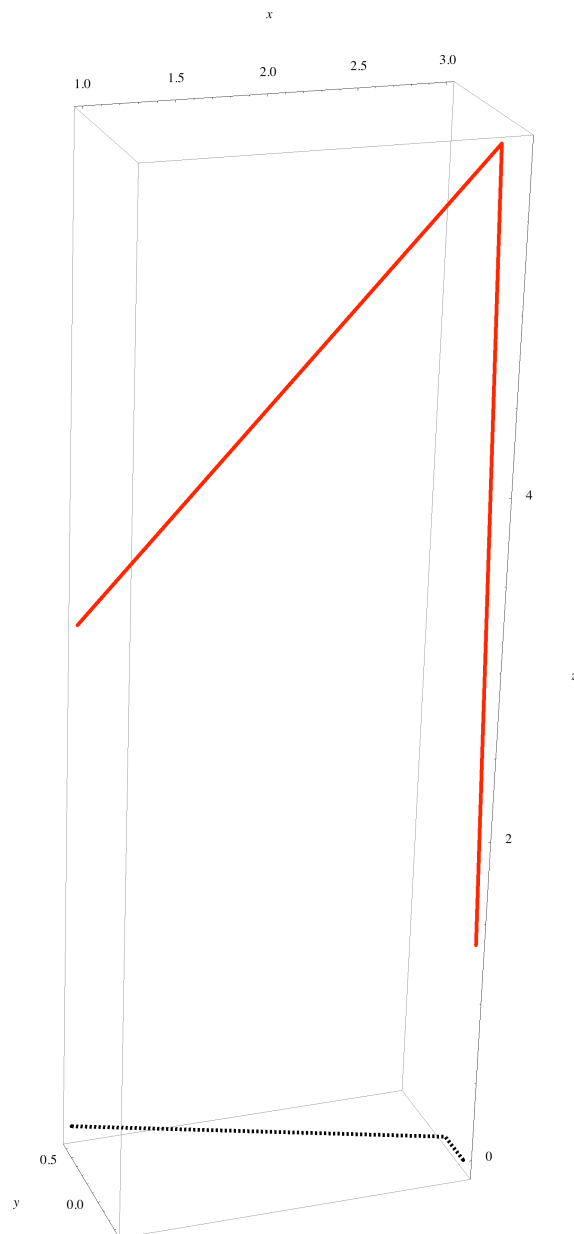
Kot med projekcijama s z komponento 0 pa je 1,21992 radianov, le tega lahko z rotacijo za $\pi/3$ okoli osi x še dodatno povečamo (slika 15). Matrika za to rotacijo izgleda tako:

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{2} & -\frac{\sqrt{3}}{2} \\ 0 & \frac{\sqrt{3}}{2} & \frac{1}{2} \end{pmatrix}$$

Vektorja sta sedaj:

$$p1_{(x \rightarrow \pi/3)} = (2, 1 - \sqrt{3}, 1 + \sqrt{3})$$

$$p2_{(x \rightarrow \pi/3)} = (0, 2 - \sqrt{3}, 1 + 2\sqrt{3})$$



Slika 15: rotacija $p1$ in $p2$ za $\pi/3$ okoli osi x

Nov kot med projekcijama pa je sedaj 1,92168 radianov.

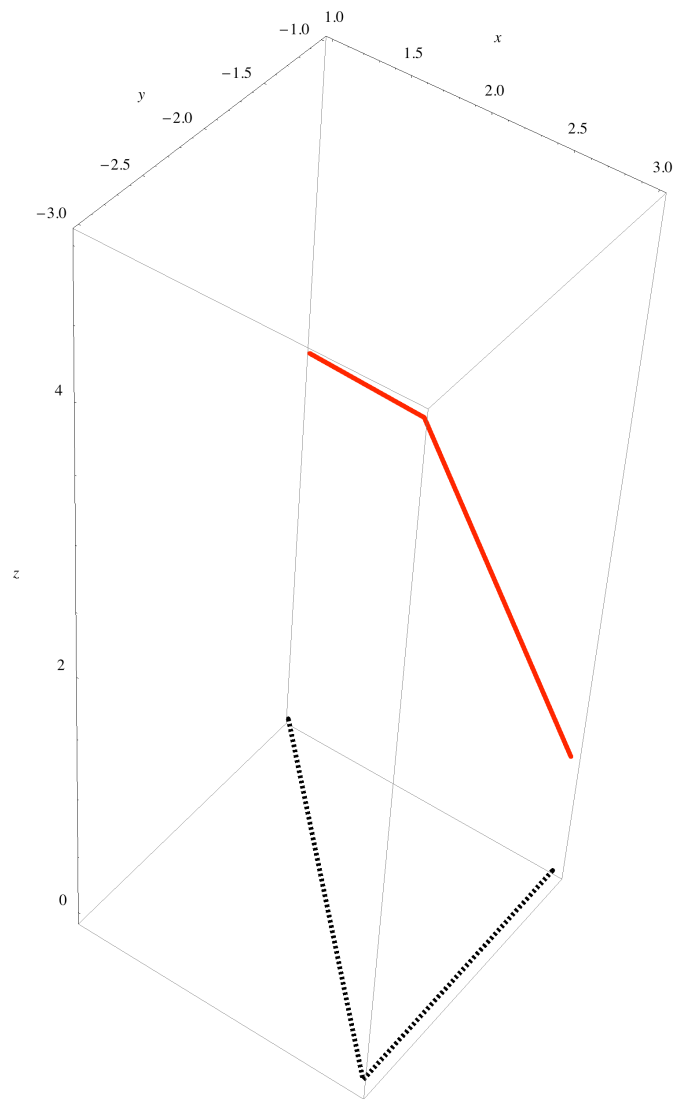
Vrtenje preko osi x za kot $\pi/2$ nam omogoča enostavna matrika:

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{pmatrix}$$

Z dano matriko dobimo vektorja:

$$p1_{(x \rightarrow \pi/2)} = (2, -2, 2)$$

$$p2_{(x \rightarrow \pi/2)} = (0, -2, 4)$$



Slika 16: rotacija p_1 in p_2 za $\pi/2$ okoli osi x

Projekciji rotiranih vektorjev p_1 in p_2 na zgornji sliki (slika 16) oklepata kot 0,785398 radianov. Zadnjo rotacijo preko osi x naredimo z matriko:

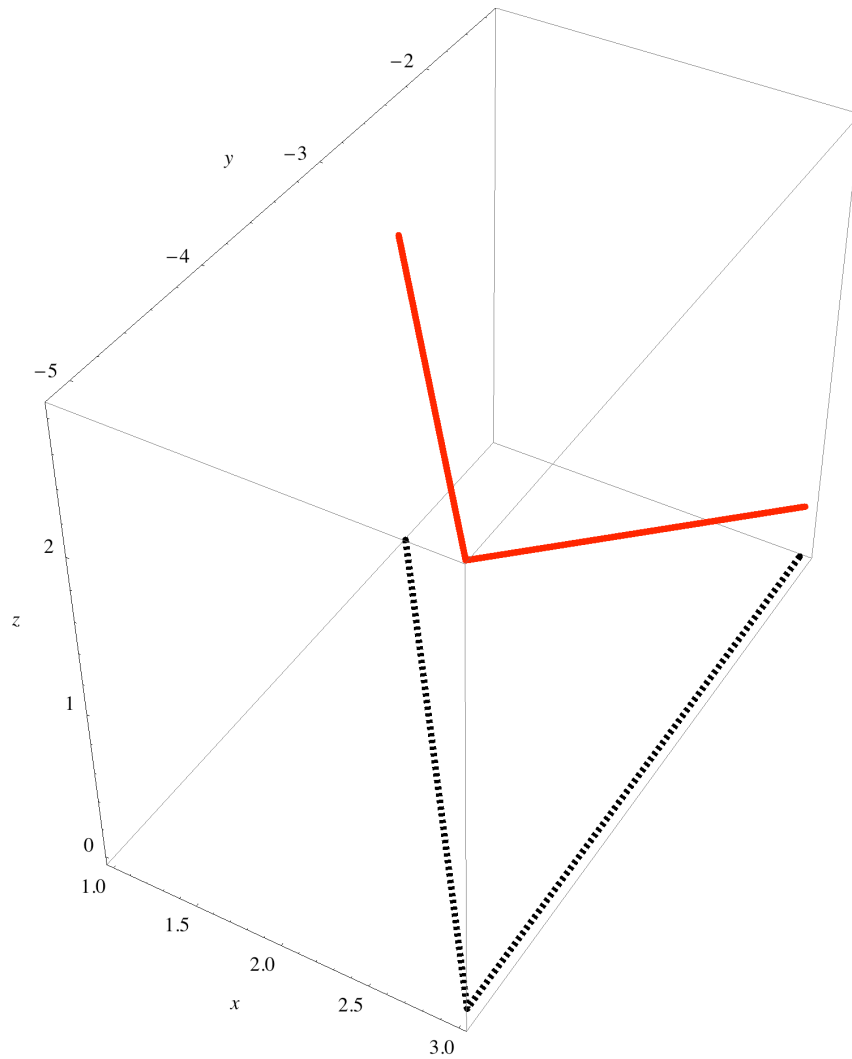
$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & -\frac{1}{2} & -\frac{\sqrt{3}}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{1}{2} \end{pmatrix}$$

Slednja rotira prvotna vektorja p_1 in p_2 oziroma prvotne točke T_1 , T_2 in T_3 za kot $2\pi/3$ okoli osi x (slika 17). Vektorja, ki jih tako dobimo sta:

$$p_{1(x \rightarrow 2\pi/3)} = (2, -1 - \sqrt{3}, -1 + \sqrt{3})$$

$$p_{2(x \rightarrow 2\pi/3)} = (0, -2 - \sqrt{3}, -1 + 2\sqrt{3})$$

Njuni projekciji se sečeta pod kotom 0,631914 radianov.



Slika 17: rotacija p1 in p2 za $2\pi/3$ okoli osi x

Tabela 1: koti projekcij

kot rotacije v radianih	kot projekcij p1 in p2	
	rotacija okoli x	rotacija okoli y
0	0.785398	0.785398
$\pi/6$	1.21992	0.693903
$\pi/3$	1.92168	0.530244
$\pi/2$	0.785398	0.321751
$2\pi/3$	0.631914	0.0577584

*kot med vektorjema p1 in p2 je v vseh primerih 0.684719 radianov

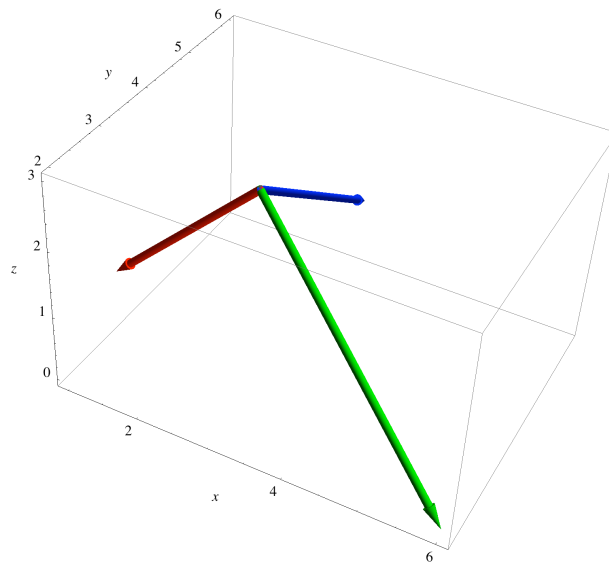
4 REKONSTRUKCIJA RAZCEPIŠČA

4.1 RAZSTAVITEV TRIDIMENZIONALNE ROGOVILE NA PROJEKCIJE IN SESTAVA ROGOVILE IZ PROJEKCIJ

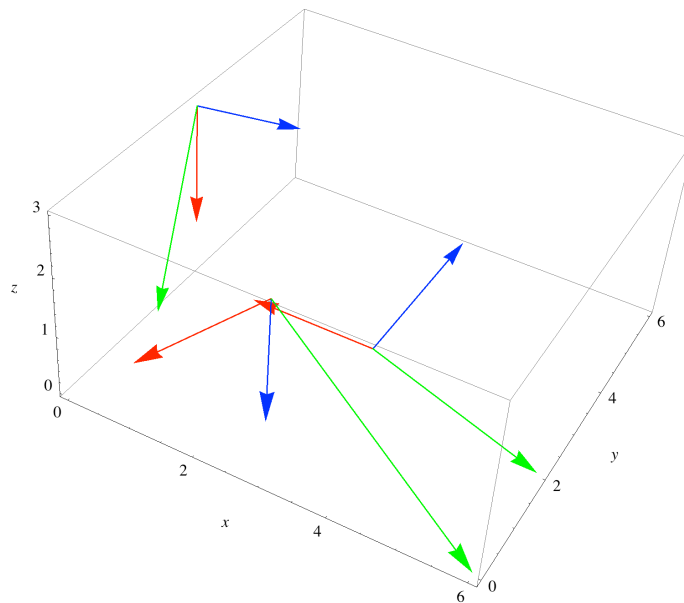
Rogovilo, ki poenostavljeno predstavlja razcepišče žil, lahko enostavno projiciramo na ortogonalne ravnine $x = 0$, $y = 0$ in $z = 0$ tako da zanemarimo x , y ali z komponento in jo nadomestimo z 0.

Primer:

Rogovilo, ki jo opisujejo trije vektorji z izhodiščem v $S(3,3,3)$ in konci v $A(3,6,1)$, $B(1,3,1)$ ter $C(6,2,0)$ razstavimo tako kot je prikazano na slikah 17 in 18.



Slika 17: prikaz rogovile, z modro barvo je označen vektor SA, z rdečo vektor SB, zeleno pa je označen vektor SC



Slika 18: ortogonalne projekcije rogovile na ravnine $x = 0$, $y = 0$ in $z = 0$

Rogovilo lahko iz projekcij ponovno sestavimo tako, da poiščemo presečišča različnih projekcij iste točke. Točka $A(3,6,1)$ se na $z = 0$ ravnini preslika v točko $A_{xy}(3,6,0)$ na $y = 0$ ravnini v točko $A_{xz}(3,0,1)$ ter na $x = 0$ ravnini v točko $A_{yz}(0,6,1)$. Normalni vektorji, ki opisujejo ravnine projekcij so $n_{xy} (0,0,1)$, $n_{xz} (0,1,0)$ in $n_{yz} (1,0,0)$. Z normalnim vektorjem ravnine projekcije in točko na njej lahko opišemo premico. Presečišče premic, ki predstavljata različni projekciji iste točke nam vrne prvotno točko.

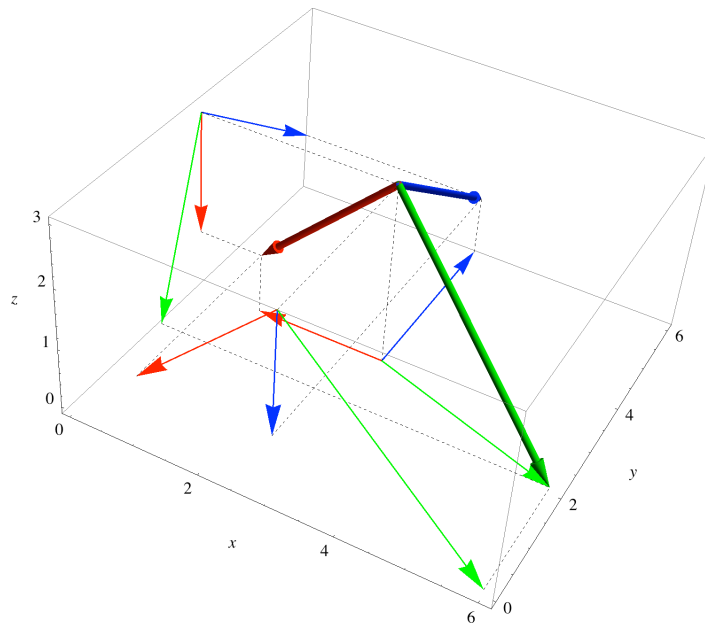
premica1: premica2:

$$x = 3 \quad x = 3$$

$$y = 6 \quad y = 0 + t_2$$

$$z = 0 + t_1 \quad z = 1$$

Z izenačitvijo zgornjih enačb je takoj razvidno da je $t_2=6$ ter $t_1=1$, kar nam da točko $X(3,6,1)$. Celoten postopek, je prikazan na spodnji sliki:



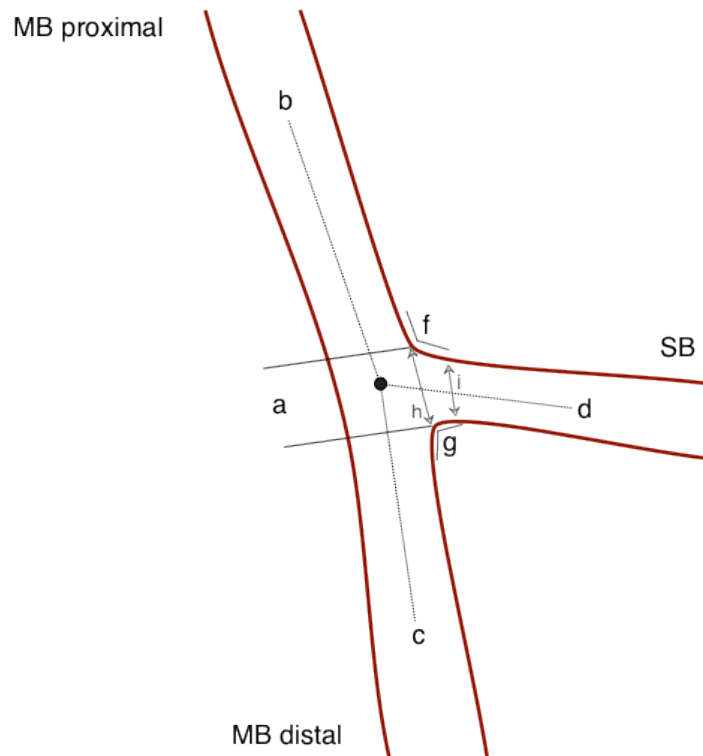
Slika 19: rogovila, njene projekcije na ortogonalne ravnine ter presečišča premic, ki predstavljajo projekcije iste točke

Koda (v jeziku Mathematica), ki izračuna presečišče dveh daljic (premico lahko simuliramo):

```
Intersections[Line[{p1_, q1_}], Line[{p2_, q2_}]] := Module[
{
v1=#/Sqrt[#.#]&[q1-p1],
v2=#/Sqrt[#.#]&[q2-p2],
v12, s1, s2
},
v12=Cross[v1, v2];
s1=Det[{p2-p1, v2, v12}]/v12.v12;
s2=Det[{p2-p1, v1, v12}]/v12.v12;
(p1+p2+v1 s1+v2 s2)/2
] /; Union[Length/@{p1, q1, p2, q2}]=={3}
```

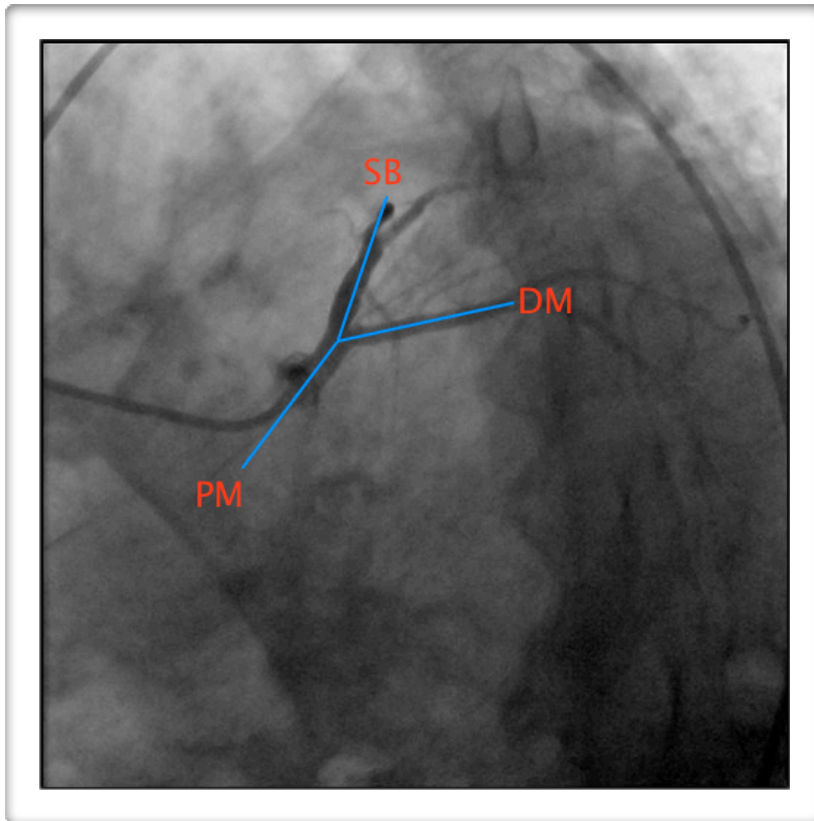

4.2 SESTAVLJANJE PROJEKCIJ KORONARK V TRIDIMENZIONALNI MODEL

Slike koronarnih arterij so pravzaprav projekcije svetline arterij na ravnine opisane s položajem detektorja rentgenskih žarkov. Položaj detektorja označujemo z dvema kotoma, ki nam povesta odklon v kranialno - kaudalni smeri (torej odklon v smeri glava - noge) ter lateralni smeri (torej odklon v levo ali desno smer). Na slikah lahko enostavno označimo poljubno razcepišče (na risbi označeno s piko), ter smer žil, ki gredo v razcepišče (proximal main - PM) in iz njega (distal main - DM ter side branch - SB).

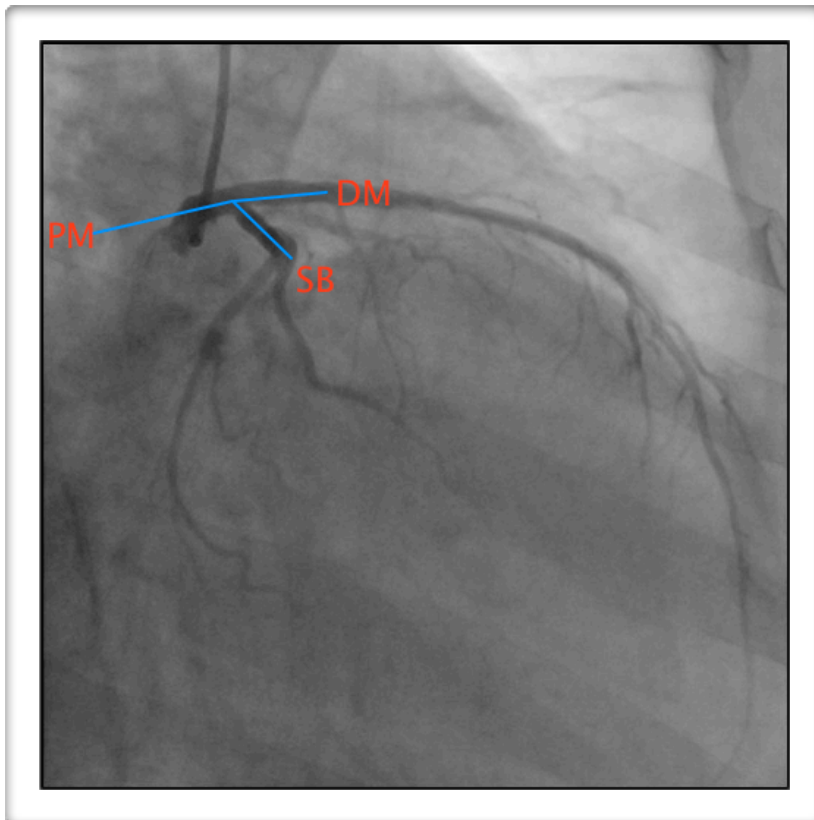


Slika 21: prikaz razcepišča proksimalne koronarne arterije (MB proximal) v stransko vejo (SB) in distalno koronarno arterijo (MB distal) (kri teče od MB proximal proti MB distal in SB)

Na zgornji sliki (slika 21) je središče razcepišča označeno s piko, smer žil pa s črtami b, d in c; f označuje kote med MB proximal (proksimalno koronarno arterijo) in SB (stranska veja), g pa kot med SB (stranska veja) ter MB distal (distalno koronarno arterijo). Ker lahko označimo presečišče kot točko, lahko z metodo sestave rogovile iz projekcij to točko poiščemo tudi v prostoru, pri tem pa moramo paziti na morebitne napake zaradi premika srca, rastrskega opisa slike, ter zamika v položaju med centrom sfere, ki jo opisuje premikanje detektorja, ter odmika lege srca od le tega. V praksi je detektor približno enako oddaljen od srca (torej je srce blizu centra sfere, ki jo premikanje detektorja opisuje), tako da je ta napaka vidna le kot majhna zamaknitev srca v določeno smer. Napako, ki jo povzroči lega detektorja odpravimo s pomočjo korekcijskih vektorjev, le-te bomo natančneje opisali v nadaljevanju. Na sliki 22 in 23 smo označili središče razcepišča in segmente PM (proksimalna arterija) SB (stranska veja), DM (distalna arterija).



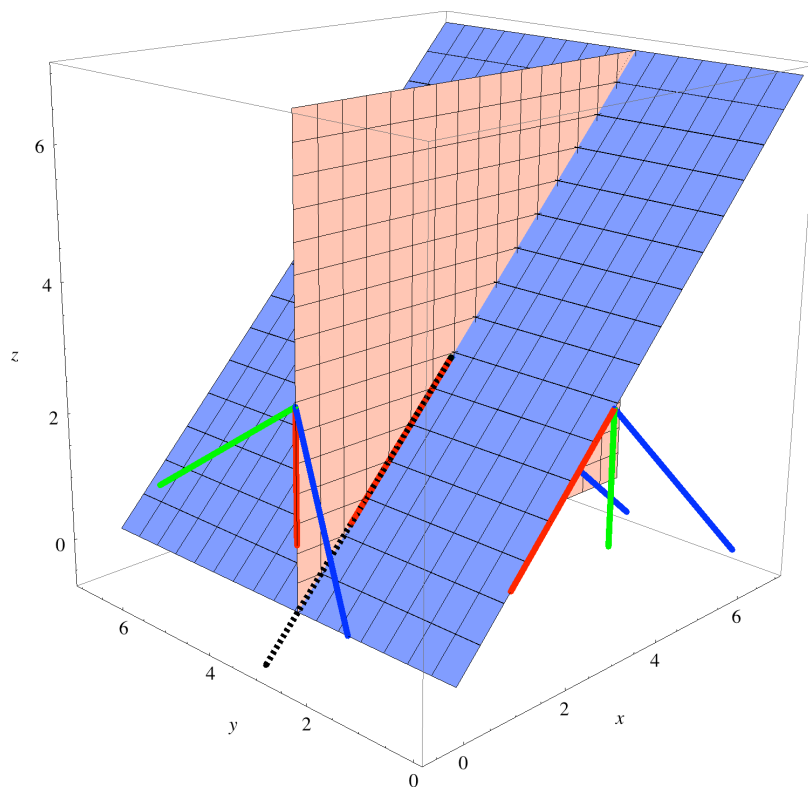
Slika 22: razcepišče koronarne žile slikano pod koti $\alpha = 47^\circ$ $\beta = 29^\circ$



Slika 23: razcepišče koronarne žile slikano pod koti $\alpha = -35^\circ$ $\beta = 26^\circ$

Napako, ki jo povzroči bitje srca zmanjšamo tako, da vzamemo tisto rentgensko sliko, ki prikazuje maksimalno napolnjeno srce (tik pred iztisolom). Problem, ki se pojavi je določitev preostalih točk, saj imamo zgolj smeri, ne pa tudi njihove dolžine. Izkaže se, da so smeri dovolj za rekonstrukcijo razcepišča, ki si ga bomo zaradi poenostavitve predstavljali kar kot rogovilo, ki jo opisujejo trije vektorji s središčem v točki S (razcepišče). Dve metodi za sestavo rogovile iz projekcij smo že spoznali, lahko si pomagamo z iskanjem presečišča normal, ki izhajajo iz različnih projekcij iste točke (slika 19) ali pa s seštevanjem vektorjev (slika 20) vendar pri tej metodi nujno potrebujemo poleg smeri vektorja, ki izhaja iz točke S , tudi njegovo velikost. Nova metoda, ki problem zaobide, pa išče namesto presečišča premic definiranih s koordinato točke projekcije in normalo ravnine na kateri točka leži, presečišče ravnin opisanih s tremi točkami, točko S , projekcijo le-te S' na dani ravnini ter poljubno drugo točko X' , ki prikazuje smer vektorja $X'S'$.

Na spodnji sliki (slika 24) je prikazan primer, kjer je rogovila z izhodiščem v $S(3,3,3)$ in koncih v $A(1,3,1)$, $B(3,6,1)$ ter $C(6,2,0)$ projicirana na $y = 0$ ter $x = 0$ ploskev. Rdeče črte prikazujejo daljico, ki se začne v točki S ter konča v točki B in njuni projekciji. Modro ravnino lahko opišemo s točko S , $S_{yz}(0,3,3)$ ter poljubno točko, ki leži na premici definirani s točko S_{yz} ter smernim vektorjem $SA_{yz} = S_{yz}(0,3,3) - A_{yz}(0,3,1) = (0,0,2)$. Oranžno ravnino prav tako opišemo s točko S (le-ta bo skupna vsem ravninam) ter točko $S_{xz}(3,0,3)$ in točko, ki leži na premici opisani z vektorjem $SB_{xz} = S_{xz}(3,0,3) - A_{xz}(1,0,1) = (2,0,2)$ ter točko S_{xz} .



Slika 24: rekonstrukcija stranice rogovile s presečiščem dveh ploskev, črtkana črna črta prikazuje rezultat

Črtkana črna črta na grafu nam predstavlja polpremico definirano s presečiščem modre in oranžne ravnine in izhodiščem v točki S .

Formulo za izračun enačbe ravnine, kadar imamo tri nekolinearne točke, ki ležijo na ravnini lahko izpeljemo iz spodnjih enačb:

$$A_1x + A_2y + A_3z - A = 0 \quad (15)$$

$$A \equiv \det \begin{pmatrix} x_1 & x_2 & x_3 \end{pmatrix} \quad (16)$$

A_i pa dobimo tako, da stolpec x_i zamenjamo s stolpcem enic (v zgornjem primeru predstavlja vsak x_i vektor).

$$\det \begin{pmatrix} 1 & y_1 & z_1 \\ 1 & y_2 & z_2 \\ 1 & y_3 & z_3 \end{pmatrix} x + \det \begin{pmatrix} x_1 & 1 & z_1 \\ x_2 & 1 & z_2 \\ x_3 & 1 & z_3 \end{pmatrix} y + \det \begin{pmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{pmatrix} z - \det \begin{pmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \end{pmatrix} = 0 \quad (17)$$

Modro ravnino podano s točkami S, S_{yz} ter B_{yz} lahko torej izračunamo tako:

$$\det \begin{pmatrix} 1 & 3 & 3 \\ 1 & 3 & 1 \\ 1 & 3 & 3 \end{pmatrix} x + \det \begin{pmatrix} 3 & 1 & 3 \\ 0 & 1 & 1 \\ 0 & 1 & 3 \end{pmatrix} y + \det \begin{pmatrix} 3 & 3 & 1 \\ 0 & 3 & 1 \\ 0 & 3 & 1 \end{pmatrix} z - \det \begin{pmatrix} 3 & 3 & 3 \\ 0 & 3 & 1 \\ 0 & 3 & 3 \end{pmatrix} = 0$$

$$-18 + 6y = 0$$

Vstavitev točk, ki opisujejo oranžno ravnino v enačbo 17 nam da:

$$\det \begin{pmatrix} 1 & 3 & 3 \\ 1 & 0 & 1 \\ 1 & 0 & 3 \end{pmatrix} x + \det \begin{pmatrix} 3 & 1 & 3 \\ 1 & 1 & 1 \\ 3 & 1 & 3 \end{pmatrix} y + \det \begin{pmatrix} 3 & 3 & 1 \\ 1 & 0 & 1 \\ 3 & 0 & 1 \end{pmatrix} z - \det \begin{pmatrix} 3 & 3 & 3 \\ 1 & 0 & 1 \\ 3 & 0 & 3 \end{pmatrix} = 0$$

$$-6x + 6z = 0$$

Premica, ki jo opiše presek modre in oranžne ravnine ima smer, ki jo lahko izračunamo kot vektorski produkt normal obeh ravnin:

$$a = n_1 \times n_2 \quad (18)$$

$$a = \left(\det \begin{pmatrix} 1 & y_1 & z_1 \\ 1 & y_2 & z_2 \\ 1 & y_3 & z_3 \end{pmatrix}, \det \begin{pmatrix} x_1 & 1 & z_1 \\ x_2 & 1 & z_2 \\ x_3 & 1 & z_3 \end{pmatrix}, \det \begin{pmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{pmatrix} \right) \times \left(\det \begin{pmatrix} 1 & b_1 & c_1 \\ 1 & b_2 & c_2 \\ 1 & b_3 & c_3 \end{pmatrix}, \det \begin{pmatrix} a_1 & 1 & c_1 \\ a_2 & 1 & c_2 \\ a_3 & 1 & c_3 \end{pmatrix}, \det \begin{pmatrix} a_1 & b_1 & 1 \\ a_2 & b_2 & 1 \\ a_3 & b_3 & 1 \end{pmatrix} \right) \quad (19)$$

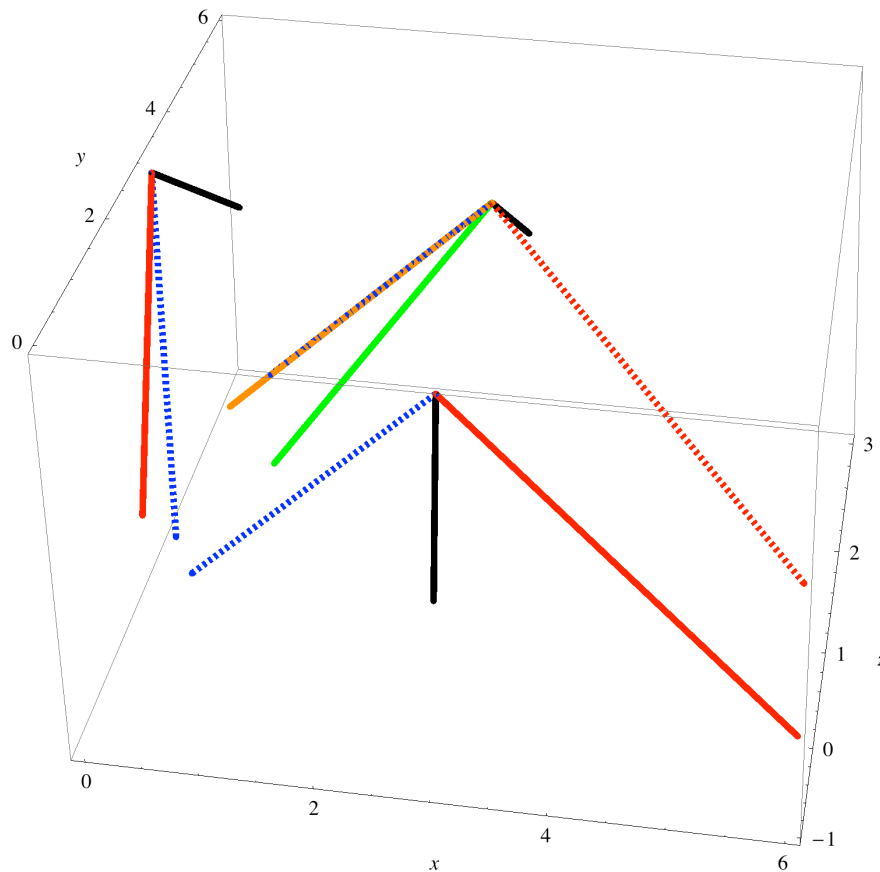
V našem primeru torej:

$$a = \left(\det \begin{pmatrix} 1 & 3 & 3 \\ 1 & 3 & 1 \\ 1 & 3 & 3 \end{pmatrix}, \det \begin{pmatrix} 3 & 1 & 3 \\ 0 & 1 & 1 \\ 0 & 1 & 3 \end{pmatrix}, \det \begin{pmatrix} 3 & 3 & 1 \\ 0 & 3 & 1 \\ 0 & 3 & 1 \end{pmatrix} \right) \times \left(\det \begin{pmatrix} 1 & 3 & 3 \\ 1 & 0 & 1 \\ 1 & 0 & 3 \end{pmatrix}, \det \begin{pmatrix} 3 & 1 & 3 \\ 1 & 1 & 1 \\ 3 & 1 & 3 \end{pmatrix}, \det \begin{pmatrix} 3 & 3 & 1 \\ 1 & 0 & 1 \\ 3 & 0 & 1 \end{pmatrix} \right)$$

$$a = (36, 0, 36)$$

Sedaj imamo smer a in točko S, tako smo na novo izgradili stranico rogovile, ki je omejena s točkama S(3,3,3) ter A(1,3,1).

Zanimivo bi bilo pogledati, kaj se zgodi, če premaknemo točko $A_{yz}(0,3,1)$ v smeri vektorja $SA_{yz}(0,0,2)$ tako, da dobimo točko $A'_{yz}(0,3,-1)$ in kako to vpliva na različne rekonstrukcijske metode. Na sliki 25 vidimo razliko med metodo s seštevanjem vektorjev in metodo s presečiščem ravnin. Zelena prikazuje rezultat, ki ga dobimo s seštevanjem vektorjev, kadar je dolžina enega vektorja projekcije zgrešena, oranžna pa prikazuje rezultat z metodo presečišča ravnin. V slednjem primeru se dobljena stranica popolnoma prilega z originalno (modra črtkana).



Slika 25: primerjava dveh metod rekonstrukcije rogovile, zelena črta predstavlja metodo seštevanja vektorjev (ta zaradi napačnega podatka o dolžini enega izmed projiciranih vektorjev odstopa), oranžna prikazuje rezultat pridobljen z metodo sekanja ravnin

4.3 ALGORITEM ZA REKONSTRUKCIJO RAZCEPIŠČA

Sedaj lahko izpeljemo celoten postopek za rekonstrukcijo žil iz koronarnih slik, na katerih je dolžina vektorjev iz razcepišča neznan in kjer je prišlo do zamika središča razcepišča. Torej podatki, ki jih imamo so:

$$\begin{aligned}
 & S_{projekcija1} \left(x_{s1}, y_{s1}, 0 \right), S_{projekcija2} \left(x_{s2}, y_{s2}, 0 \right), \dots \\
 & A_{projekcija1}^1 \left(x_{a1}, y_{a1}, 0 \right), A_{projekcija2}^2 \left(x_{a2}, y_{a2}, 0 \right), \dots \\
 & B_{projekcija1}^1 \left(x_{b1}, y_{b1}, 0 \right), B_{projekcija2}^2 \left(x_{b2}, y_{b2}, 0 \right), \dots \quad (20) \\
 & C_{projekcija1}^1 \left(x_{c1}, y_{c1}, 0 \right), A_{projekcija2}^2 \left(x_{c2}, y_{c2}, 0 \right), \dots \\
 & \alpha_{projekcija1}, \alpha_{projekcija2}, \dots \\
 & \beta_{projekcija1}, \beta_{projekcija2}, \dots
 \end{aligned}$$

Dane točke moramo zavrteti preko osi x in y za kota α in β , da dobimo točke na pravih ravninah:

$$\begin{aligned}
 S'_{projekcija_i} &= \begin{pmatrix} x_{s1}, & y_{s1}, & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha_{p_i} & -\sin\alpha_{p_i} \\ 0 & \sin\alpha_{p_i} & \cos\alpha_{p_i} \end{pmatrix} \begin{pmatrix} \cos\beta_{p_i} & 0 & \sin\beta_{p_i} \\ 0 & 1 & 0 \\ -\sin\beta_{p_i} & 0 & \cos\beta_{p_i} \end{pmatrix} \\
 A'_{projekcija_i} &= \begin{pmatrix} x_{a1}, & y_{a1}, & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha_{p_i} & -\sin\alpha_{p_i} \\ 0 & \sin\alpha_{p_i} & \cos\alpha_{p_i} \end{pmatrix} \begin{pmatrix} \cos\beta_{p_i} & 0 & \sin\beta_{p_i} \\ 0 & 1 & 0 \\ -\sin\beta_{p_i} & 0 & \cos\beta_{p_i} \end{pmatrix} \\
 B'_{projekcija_i} &= \begin{pmatrix} x_{b1}, & y_{b1}, & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha_{p_i} & -\sin\alpha_{p_i} \\ 0 & \sin\alpha_{p_i} & \cos\alpha_{p_i} \end{pmatrix} \begin{pmatrix} \cos\beta_{p_i} & 0 & \sin\beta_{p_i} \\ 0 & 1 & 0 \\ -\sin\beta_{p_i} & 0 & \cos\beta_{p_i} \end{pmatrix} \\
 C'_{projekcija_i} &= \begin{pmatrix} x_{c1}, & y_{c1}, & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha_{p_i} & -\sin\alpha_{p_i} \\ 0 & \sin\alpha_{p_i} & \cos\alpha_{p_i} \end{pmatrix} \begin{pmatrix} \cos\beta_{p_i} & 0 & \sin\beta_{p_i} \\ 0 & 1 & 0 \\ -\sin\beta_{p_i} & 0 & \cos\beta_{p_i} \end{pmatrix}
 \end{aligned} \tag{21}$$

Nato ustvarimo dve premici, ki imata smer normale ravnin ustvarjenih s kotoma α in β ter izvorom v točki S':

$$\begin{aligned}
 n_{\alpha_1\beta_1} &= \begin{pmatrix} 1, & 1, & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha_{p1} & -\sin\alpha_{p1} \\ 0 & \sin\alpha_{p1} & \cos\alpha_{p1} \end{pmatrix} \begin{pmatrix} \cos\beta_{p1} & 0 & \sin\beta_{p1} \\ 0 & 1 & 0 \\ -\sin\beta_{p1} & 0 & \cos\beta_{p1} \end{pmatrix} \\
 n_{\alpha_2\beta_2} &= \begin{pmatrix} 1, & 1, & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha_{p2} & -\sin\alpha_{p2} \\ 0 & \sin\alpha_{p2} & \cos\alpha_{p2} \end{pmatrix} \begin{pmatrix} \cos\beta_{p2} & 0 & \sin\beta_{p2} \\ 0 & 1 & 0 \\ -\sin\beta_{p2} & 0 & \cos\beta_{p2} \end{pmatrix} \tag{22}
 \end{aligned}$$

⋮

$$P_1 = S'_{projekcija1} + n_{\alpha_1\beta_1}\phi_1$$

$$P_2 = S'_{projekcija2} + n_{\alpha_2\beta_2}\phi_2$$

⋮

Preverimo ali imata dobljeni premici presečišče (zaradi rastrskega značaja koronarnih slik je to malo verjetno), če sta mimobežnici mora veljati naslednje:

$$\begin{aligned}
S'_{projekcija1} &= \begin{pmatrix} x'_{s1}, & y'_{s1}, & z'_{s1} \end{pmatrix} \\
S'_{projekcija2} &= \begin{pmatrix} x'_{s2}, & y'_{s2}, & z'_{s2} \end{pmatrix} \\
&\vdots \\
n_{\alpha_1\beta_1} &= \begin{pmatrix} a_{\alpha_1\beta_1}, & b_{\alpha_1\beta_1}, & c_{\alpha_1\beta_1} \end{pmatrix} \quad (23) \\
n_{\alpha_2\beta_2} &= \begin{pmatrix} a_{\alpha_2\beta_2}, & b_{\alpha_2\beta_2}, & c_{\alpha_2\beta_2} \end{pmatrix} \\
&\vdots \\
&\left| \begin{array}{ccc|c}
x'_{s1} & y'_{s1} & z'_{s1} & 1 \\
a_{\alpha_1\beta_1} & b_{\alpha_1\beta_1} & c_{\alpha_1\beta_1} & 1 \\
x'_{s2} & y'_{s2} & z'_{s2} & 1 \\
a_{\alpha_2\beta_2} & b_{\alpha_2\beta_2} & c_{\alpha_2\beta_2} & 1
\end{array} \right| \neq 0
\end{aligned}$$

Sicer pa takoj poiščemo točko presečišča z izenačitvijo enačb in tako dobimo točko S v naši rekonstrukciji razcepišča žil.

V kolikor se premici ne sečeta moramo poiskati najkrajšo razdaljo med njima in smer, ki je pravokotna na obe, tako bomo dobili točki, ki sta si med seboj najbližji. Aritmetična sredina slednjih nam bo dala točko S, ki jo iščemo.

$$\begin{aligned}
smer &= \frac{n_{\alpha_1\beta_1} \times n_{\alpha_2\beta_2}}{|n_{\alpha_1\beta_1} \times n_{\alpha_2\beta_2}|} \\
&\pm \begin{vmatrix} x'_{s1} - x'_{s2} & y'_{s1} - y'_{s2} & z'_{s1} - z'_{s2} \\ a_{\alpha_1\beta_1} & b_{\alpha_1\beta_1} & c_{\alpha_1\beta_1} \\ a_{\alpha_2\beta_2} & b_{\alpha_2\beta_2} & c_{\alpha_2\beta_2} \end{vmatrix} \quad (24) \\
razdalja &= \frac{\pm \begin{vmatrix} x'_{s1} - x'_{s2} & y'_{s1} - y'_{s2} & z'_{s1} - z'_{s2} \\ a_{\alpha_1\beta_1} & b_{\alpha_1\beta_1} & c_{\alpha_1\beta_1} \\ a_{\alpha_2\beta_2} & b_{\alpha_2\beta_2} & c_{\alpha_2\beta_2} \end{vmatrix}}{\sqrt{\begin{vmatrix} a_{\alpha_1\beta_1} & b_{\alpha_1\beta_1} \\ a_{\alpha_2\beta_2} & b_{\alpha_2\beta_2} \end{vmatrix}^2 + \begin{vmatrix} b_{\alpha_1\beta_1} & c_{\alpha_1\beta_1} \\ b_{\alpha_2\beta_2} & c_{\alpha_2\beta_2} \end{vmatrix}^2 + \begin{vmatrix} c_{\alpha_1\beta_1} & a_{\alpha_1\beta_1} \\ c_{\alpha_2\beta_2} & b_{\alpha_2\beta_2} \end{vmatrix}^2}}
\end{aligned}$$

Premico P_1 sedaj lahko premaknemo in dobimo:

$$P'_1 = smer * razdalja + S'_{projekcija1} + n_{\alpha_1\beta_1} \phi_1 \quad (25)$$

To sedaj enačimo s P_2 in dobimo presečišče tako da iz sistema enačb izračunamo spremenljivki Φ_1 in Φ_2 .

$$smer = \begin{pmatrix} s_x, & s_y, & s_z \end{pmatrix}$$

$$P'_1 x = s_x * razdalja + x'_{s1} + a_{\alpha_1\beta_1} \phi_1 = x'_{s2} + a_{\alpha_2\beta_2} \phi_2 = P_2 \quad (26)$$

$$P'_1 y = s_y * razdalja + y'_{s1} + b_{\alpha_1\beta_1} \phi_1 = y'_{s2} + b_{\alpha_2\beta_2} \phi_2 = P_2$$

$$P'_1 z = s_z * razdalja + z'_{s1} + c_{\alpha_1\beta_1} \phi_1 = z'_{s2} + c_{\alpha_2\beta_2} \phi_2 = P_2$$

Primer rešitve sistema:

$$\phi_2 = \frac{s_x * razdalja + x'_{s1} + a_{\alpha_1\beta_1}\phi_1 - x'_{s2}}{a_{\alpha_2\beta_2}}$$

$$\phi_1 = \frac{y'_{s2} + b_{\alpha_2\beta_2}\phi_2 - s_y * razdalja - y'_{s1}}{b_{\alpha_1\beta_1}}$$

$$\phi_1 = \frac{y'_{s2} + b_{\alpha_2\beta_2} \frac{s_x * razdalja + x'_{s1} + a_{\alpha_1\beta_1}\phi_1 - x'_{s2}}{a_{\alpha_2\beta_2}} - s_y * razdalja - y'_{s1}}{b_{\alpha_1\beta_1}}$$

$$\phi_1 b_{\alpha_1\beta_1} = y'_{s2} + b_{\alpha_2\beta_2} \frac{s_x * razdalja + x'_{s1} + a_{\alpha_1\beta_1}\phi_1 - x'_{s2}}{a_{\alpha_2\beta_2}} - s_y * razdalja - y'_{s1}$$

$$\phi_1 b_{\alpha_1\beta_1} a_{\alpha_2\beta_2} = a_{\alpha_2\beta_2} y'_{s2} + b_{\alpha_2\beta_2} (s_x * razdalja + x'_{s1} + a_{\alpha_1\beta_1}\phi_1 - x'_{s2}) - a_{\alpha_2\beta_2} s_y * razdalja - a_{\alpha_2\beta_2} y'_{s1}$$

$$\phi_1 b_{\alpha_1\beta_1} a_{\alpha_2\beta_2} - b_{\alpha_2\beta_2} a_{\alpha_1\beta_1} \phi_1 = a_{\alpha_2\beta_2} y'_{s2} + b_{\alpha_2\beta_2} s_x * razdalja + b_{\alpha_2\beta_2} x'_{s1} - b_{\alpha_2\beta_2} x'_{s2} - a_{\alpha_2\beta_2} s_y * razdalja - a_{\alpha_2\beta_2} y'_{s1}$$

$$\phi_1 = \frac{a_{\alpha_2\beta_2} y'_{s2} + b_{\alpha_2\beta_2} s_x * razdalja + b_{\alpha_2\beta_2} x'_{s1} - b_{\alpha_2\beta_2} x'_{s2} - a_{\alpha_2\beta_2} s_y * razdalja - a_{\alpha_2\beta_2} y'_{s1}}{b_{\alpha_1\beta_1} a_{\alpha_2\beta_2} - b_{\alpha_2\beta_2} a_{\alpha_1\beta_1}}$$

(27)

Iz dobljenih točk T₁ in T₂ izračunamo aritmetično sredino in dobimo točko S. Natančnejše rezultate lahko dobimo z uporabo večih projekcij, v tem primeru je število točk, ki nas zanimajo pri rekonstrukciji središča razcepišča (S) odvisno od števila uporabljenih projekcij, odvisnost je prikazana v enačbi 28. S dobimo tako da dobljene točke vstavimo v enačbo 12.

$$f_{(2)} = 2$$

$$f_{(n)} = 2(n-1) + f_{(n-1)} \quad (28)$$

Dobili smo središče razcepišča koronark. Sedaj moramo dobljeno središče ponovno projicirati na že znane ravnine, da lahko izračunamo korekcijske vektorje, s katerimi bomo popravili zamike točk. Projiciramo ga tako da sledimo smeri normalnih vektorjev ravnin, ta metoda je uporabna tudi pri sledenju žarkov. Za izračun presečišča lahko uporabimo enačbo 29 ali 30.

$$S = \begin{pmatrix} x_s, & y_s, & z_s \end{pmatrix}$$

$$n_{\alpha_1\beta_1} = \begin{pmatrix} a_{\alpha_1\beta_1}, & b_{\alpha_1\beta_1}, & c_{\alpha_1\beta_1} \end{pmatrix}$$

$$S_{projekcija1}^k = \begin{pmatrix} a_{\alpha_1\beta_1} & B'_{projekcija1}x - A'_{projekcija1}x & C'_{projekcija1}x - A'_{projekcija1}x \\ b_{\alpha_1\beta_1} & B'_{projekcija1}y - A'_{projekcija1}y & C'_{projekcija1}y - A'_{projekcija1}y \\ c_{\alpha_1\beta_1} & B'_{projekcija1}z - A'_{projekcija1}z & C'_{projekcija1}z - A'_{projekcija1}z \end{pmatrix}^{-1} \begin{pmatrix} x_s - A'_{projekcija1}x \\ y_s - A'_{projekcija1}y \\ z_s - A'_{projekcija1}z \end{pmatrix} \quad (29)$$

⋮

Enostavnejši izračun presečišča vendar računalniško počasnejši je tudi:

$$\text{Premica: } \frac{x - x_s}{a_{\alpha_1\beta_1}} = \frac{y - y_s}{b_{\alpha_1\beta_1}} = \frac{z - z_s}{c_{\alpha_1\beta_1}}$$

$$\text{Ravnina: } Ax + By + Cz + D = 0$$

$$x_p = x_s - a_{\alpha_1\beta_1} p \quad (30)$$

$$y_p = y_s - b_{\alpha_1\beta_1} p$$

$$z_p = z_s - c_{\alpha_1\beta_1} p$$

$$p = \frac{Ax_s + By_s + Cz_s + D}{Aa_{\alpha_1\beta_1} + Bb_{\alpha_1\beta_1} + Cc_{\alpha_1\beta_1}}$$

Korekcijske vektorje dobimo kot smo že opisali z enačbo 31:

$$\text{korekcijskiVektor}_i = S_{projekcija_i}^k - S'_{projekcija_i} \quad (31)$$

Vse točke sedaj popravimo:

$$X_{projekcija_i}^k = X'_{projekcija_i} + \text{korekcijskiVektor} \quad (32)$$

S pridobljenimi podatki lahko izračunamo normale ravnin sestavljenih iz točk:

$$S, \quad S_{projekcija_i}^k, \quad X_{projekcija_i}^k$$

(X je katerakoli izmed točk A, B ali C) s pomočjo formule:

$$Ax + By + Cz - D = 0$$

$$A = \begin{vmatrix} 1 & y_1 & z_1 \\ 1 & y_2 & z_2 \\ 1 & y_3 & z_3 \end{vmatrix}$$

$$B = \begin{vmatrix} x_1 & 1 & z_1 \\ x_2 & 1 & z_2 \\ x_3 & 1 & z_3 \end{vmatrix} \quad (33)$$

$$C = \begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix}$$

Smer vektorjev, ki gredo iz točke S in opisujejo smeri žil, ki potekajo iz razcepišča nato enostavno izračunamo z vektorskim produktom normal ravnin, ki predstavljajo iste toče v različnih projekcijah:

$$\text{smer_koronarke}_x = \frac{\left(\begin{matrix} A_i & B_i & C_i \end{matrix} \right) \times \left(\begin{matrix} A_j & B_j & C_j \end{matrix} \right)}{\left\| \left(\begin{matrix} A_i & B_i & C_i \end{matrix} \right) \times \left(\begin{matrix} A_j & B_j & C_j \end{matrix} \right) \right\|} \quad (34)$$

Na tem mestu je treba poudariti, da eno izmed ravnin sestavljajo točke S, $S_{projekcija(i)}$ ter ena izmed točk A, B ali C v projekciji (i), to točko bomo označili kot $X_{projekcija(i)}$, drugo pa točke S, $S_{projekcija(k)}$ ter $X_{projekcija(k)}$. Kadar uporabljamo več kot dve projekciji za rekonstrukcijo razcepišča, dobimo več možnih smeri koronarke_x, v tem primeru izračunamo vse možne smeri, katerih število je podano z enačbo 34, ter jih seštejemo.

$$g_{(2)} = 1$$

$$g_{(x)} = (x-1) + g_{(x-1)} \quad (35)$$

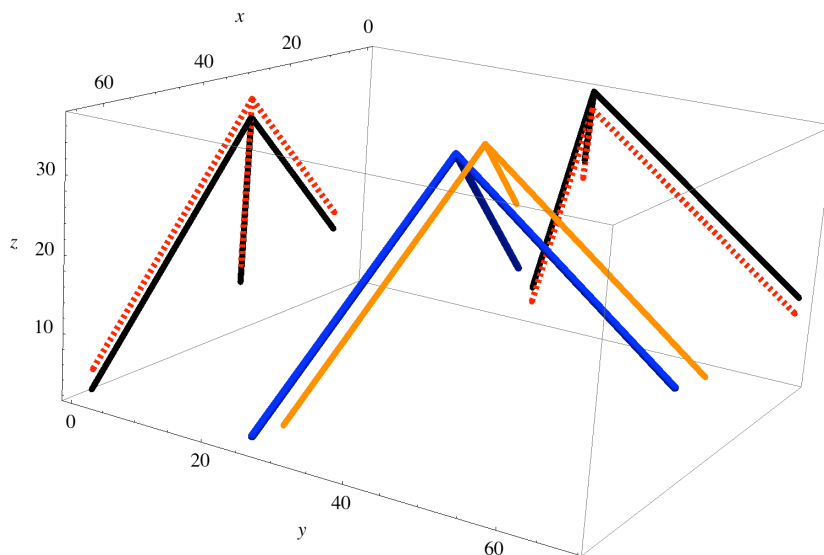
Problem, ki nastane pri izračunu smeri koronarke_x (enačba 34) je ta, da ne vemo ali naj to smer točki S prištejemo ali odštejemo za rekonstrukcijo koronarke_x. To rešimo tako, da s pomočjo enačbe 30 poiščemo presečišče premice, definirane s točko S in smernim vektorjem *smer_koronarke_x*, ter ravnine definirane s točko X_{projekcija(i)} in normalnim vektorjem $n_i = S_{\text{projekcija}(i)} - X_{\text{projekcija}(i)}$. Presečišče nam predstavlja rekonstruirano točko X.

Na koncu lahko izračunamo še kot med koronarkami s pomočjo enačbe 36:

$$\mu \cdot v = \|\mu\| \|v\| \cos \theta$$

$$\cos \theta = \frac{\mu \cdot v}{\|\mu\| \|v\|} \quad (36)$$

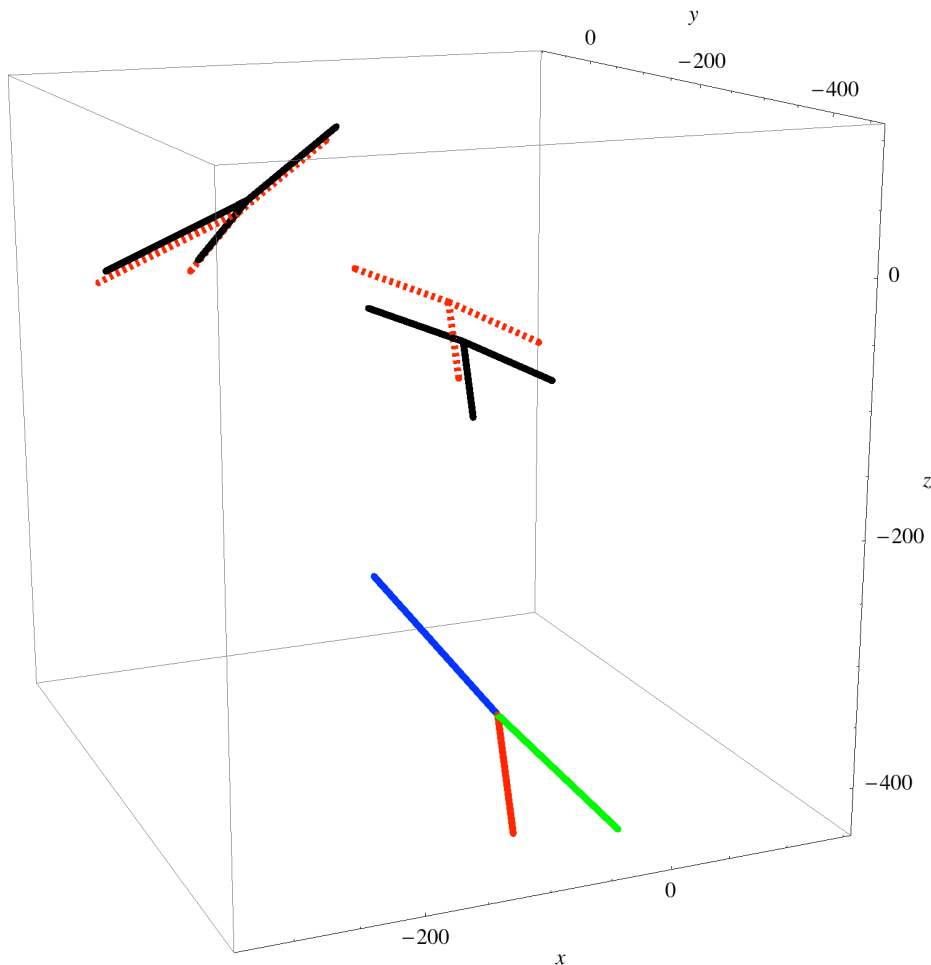
Za primer si lahko ogledamo rogovilo s središčem v točki S(33,33,33) iz katere gredo vektorji v točke A(11,32,13), B(34,66,10) ter C(65,26,3). Poleg danih točk bomo simulirali nenatančnost detektorja tako, da bomo vse projekcije zamaknili na $x = 0$ ravnini z vektorjem $n(0,3,4)$ ter na $y = 0$ ravnini z vektorjem $m(-2,0,1)$ ter podaljšali enega od vektorjev s premikom točke A' na ravnini $x = 0$ tako da bo namesto A'(0,32,13) imela vrednost A'(0,71/2,27). Po rekonstrukciji si lahko ogledamo lego originalne rogovile, naše rogovile in primerjamo kote. Na spodnji sliki (slika 26) je originalna rogovila obarvana modro, rekonstrukcija le te pa oranžno. Kljub korekciji je prišlo do majhnega zamika, z večjim številom projekcij bi se ta napaka zmanjšala. S črno barvo so označene prvotne projekcije z zamikom, rdeče pa so označene popravljene projekcije. Kot originalne rogovile med vektorjema SA in SB znaša 1.22564 radianov, kot istoležečih vektorjev na rekonstruirani rogovili je prav tako 1.22564 radianov, poleg tega kota se ujemajo tudi vsi ostali, tako da smo natančno rekonstruirali rogovilo.



Slika 26: rekonstrukcija rogovile z upoštevanjem napak zaradi lege detektorja rentgenskih žarkov (na vsaki projekciji so točke zamaknjene z vektorjem, ki simulira napako detektorja) in napak pri označevanju dolžine posameznega vektorja (eden izmed vektorjev na projekciji ima popačeno dolžino). Modre barve je originalna rogovila, ki predstavlja dejansko razcepišče žil, oranžne barve je rogovila, ki smo jo rekonstruirali iz projekcij.

5. PRIMER

Iz slik 22 in 23 smo odčitali sredino razcepišča in smer arterij v dveh projekcijah. Dobljene točke so $S'_{(slika1)}(201,204)$, $PM'_{(slika1)}(136,289)$, $DM'_{(slika1)}(320,177)$ in $SB'_{(slika1)}(234,104)$ na prvi projekciji ter $S'_{(slika2)}(129,107)$, $PM'_{(slika2)}(34,128)$, $DM'_{(slika2)}(227,94)$ in $SB'_{(slika2)}(196,173)$. Poleg tega vemo da je prva projekcija slikana pod koti $\alpha_{(1)}=47^\circ$ in $\beta_{(1)}=29^\circ$, druga pa pod koti $\alpha_{(2)}=-35^\circ$ ter $\beta_{(2)}=26^\circ$. Na sliki 27 je viden rezultat vnosa zgornjih podatkov v algoritem, ki smo ga izdelali v tej diplomski nalogi in opisali v prejšnjem poglavju. S črno barvo so prikazane originalne lege točk na projekcijah, črtkano rdeče pa lege točk po uporabi korekcijskih vektorjev. Rekonstrukcija je prikazana večbarvno, tako da lahko enostavno ločimo arterije med seboj.



Slika 27: rekonstrukcija razcepišča prikazanega na slikah 22 in 23, z modro je označena proksimalna glavna arterija (PM), z zeleno distalna glavna arterija (DM), z rdečo pa stranska veja (SB); kot med PM in DM je 167° , kot med DM in SB pa 47°

6 DEMONSTRACIJSKA KODA

6.1 IZRAČUN PROJEKCIJ IN KOTOV

```

p1 = {{1, 3, 1}, {3, 5, 3}, {3, 1, 1}}
s1 = p1; s1[[All, 3]] = 0
Graphics3D[{{Thick, Red, Line[p1]}, {Dashed, Line[s1]}}]
p1vektor1 = -p1[[1]] + p1[[2]]
p1vektor2 = +p1[[2]] - p1[[3]]
N[VectorAngle[p1vektor1, p1vektor2]]
s1vektor1 = -s1[[1]] + s1[[2]]
s1vektor2 = +s1[[2]] - s1[[3]]
N[VectorAngle[s1vektor1, s1vektor2]]
Clear[p1]
p1 = Line[{{1, 3, 1}, {3, 5, 3}, {3, 1, 1}}]
Clear[p2, s2]
RotationMatrix[Pi/6, {0, 1, 0}] // MatrixForm
p2 := Rotate[p1, Pi/6, {0, 1, 0}]
s2 = Line[{{1/2 + Sqrt[3]/2, 3, 0}, {3/2 + (3 Sqrt[3])/2, 5,
0}, {1/2 + (3 Sqrt[3])/2, 1, 0}}]
Graphics3D[{{Thick, Red, p2}, {Dashed, s2}}, Axes -> True]
Normal@p2
p2vektor1 = -{1/2 + Sqrt[3]/2, 3, -(1/2) + Sqrt[3]/2} + {3/2 +
(3 Sqrt[3])/2, 5, -(3/2) + (3 Sqrt[3])/2}
p2vektor2 = +{3/2 + (3 Sqrt[3])/2, 5, -(3/2) + (3 Sqrt[3])/2}
- {1/2 + (3 Sqrt[3])/2, 1, -(3/2) + Sqrt[3]/2}
N[VectorAngle[p2vektor1, p2vektor2]]
s2vektor1 = -{1/2 + Sqrt[3]/2, 3, 0} + {3/2 + (3 Sqrt[3])/2,
5, 0}
s2vektor2 = +{3/2 + (3 Sqrt[3])/2, 5, 0} - {1/2 + (3 Sqrt[3])/
2, 1, 0}
N[VectorAngle[s2vektor1, s2vektor2]]
RotationMatrix[Pi/3, {0, 1, 0}] // MatrixForm
p3 := Rotate[p1, Pi/3, {0, 1, 0}]
s3 = Line[{{1/2 + Sqrt[3]/2, 3, 0}, {3/2 + (3 Sqrt[3])/2, 5,
0}, {3/2 + Sqrt[3]/2, 1, 0}}]
Graphics3D[{{Thick, Red, p3}, {Dashed, s3}}, Axes -> True]
Normal@p3
p3vektor1 = -{1/2 + Sqrt[3]/2, 3, 1/2 - Sqrt[3]/2} + {3/2 + (3
Sqrt[3])/2, 5, 3/2 - (3 Sqrt[3])/2}
p3vektor2 = +{3/2 + (3 Sqrt[3])/2, 5, 3/2 - (3 Sqrt[3])/2} -
{3/2 + Sqrt[3]/2, 1, 1/2 - (3 Sqrt[3])/2}
N[VectorAngle[p3vektor1, p3vektor2]]
s3vektor1 = -{1/2 + Sqrt[3]/2, 3, 0} + {3/2 + (3 Sqrt[3])/2,
5, 0}
s3vektor2 = +{3/2 + (3 Sqrt[3])/2, 5, 0} - {3/2 + Sqrt[3]/2,
1, 0}
N[VectorAngle[s3vektor1, s3vektor2]]

```

```

RotationMatrix[Pi/2, {0, 1, 0}] // MatrixForm
p4 := Rotate[p1, Pi/2, {0, 1, 0}]
s4 = Line[{{1, 3, 0}, {3, 5, 0}, {1, 1, 0}}]
Graphics3D[{{Thick, Red, p4}, {Dashed, s4}}, Axes -> True]
Normal@p4
p4vektor1 = -{1, 3, -1} + {3, 5, -3}
p4vektor2 = +{3, 5, -3} - {1, 1, -3}
N[VectorAngle[p4vektor1, p4vektor2]]
s4vektor1 = -{1, 3, 0} + {3, 5, 0}
s4vektor2 = {3, 5, 0} - {1, 1, 0}
N[VectorAngle[s4vektor1, s4vektor2]]
RotationMatrix[2 Pi/3, {0, 1, 0}] // MatrixForm
p5 := Rotate[p1, 2 Pi/3, {0, 1, 0}]
s5 = Line[{{-(1/2) + Sqrt[3]/2, 3, 0}, {-(3/2) + (3 Sqrt[3])/
2, 5, 0}, {-(3/2) + Sqrt[3]/2, 1, 0}}]
Graphics3D[{{Thick, Red, p5}, {Dashed, s5}}, Axes -> True]
Normal@p5
p5vektor1 = -{-(1/2) + Sqrt[3]/2, 3, -(1/2) - Sqrt[3]/2} + {-
(3/2) + (3 Sqrt[3])/2, 5, -(3/2) - (3 Sqrt[3])/2}
p5vektor2 = +{-(3/2) + (3 Sqrt[3])/2, 5, -(3/2) - (3 Sqrt[3])/
2} - {-(3/2) + Sqrt[3]/2, 1, -(1/2) - (3 Sqrt[3])/2}
N[VectorAngle[p5vektor1, p5vektor2]]
s5vektor1 = -{-(1/2) + Sqrt[3]/2, 3, 0} + {-(3/2) + (3
Sqrt[3])/2, 5, 0}
s5vektor2 = +{-(3/2) + (3 Sqrt[3])/2, 5, 0} - {-(3/2) +
Sqrt[3]/2, 1, 0}
N[VectorAngle[s5vektor1, s5vektor2]]
(*rotacija okoli x*)
RotationMatrix[Pi/6, {1, 0, 0}] // MatrixForm
xp2 := Rotate[p1, Pi/6, {1, 0, 0}]
xs2 = Line[{{1, -(1/2) + (3 Sqrt[3])/2, 0}, {3, -(3/2) + (5
Sqrt[3])/2, 0}, {3, -(1/2) + Sqrt[3]/2, 0}}]
Graphics3D[{{Thick, Red, xp2}, {Dashed, xs2}}, Axes -> True]
Normal@xp2
xp2vektor1 = -{1, -(1/2) + (3 Sqrt[3])/2, 3/2 + Sqrt[3]/2} +
{3, -(3/2) + (5 Sqrt[3])/2, 5/2 + (3 Sqrt[3])/2}
xp2vektor2 = +{3, -(3/2) + (5 Sqrt[3])/2, 5/2 + (3 Sqrt[3])/2}
- {3, -(1/2) + Sqrt[3]/2, 1/2 + Sqrt[3]/2}
N[VectorAngle[xp2vektor1, xp2vektor2]]
xs2vektor1 = -{1, -(1/2) + (3 Sqrt[3])/2, 0} + {3, -(3/2) + (5
Sqrt[3])/2, 0}
xs2vektor2 = +{3, -(3/2) + (5 Sqrt[3])/2, 0} - {3, -(1/2) +
Sqrt[3]/2, 0}
N[VectorAngle[xs2vektor1, xs2vektor2]]
RotationMatrix[Pi/3, {1, 0, 0}] // MatrixForm
xp3 := Rotate[p1, Pi/3, {1, 0, 0}]
xs3 = Line[{{1, 3/2 - Sqrt[3]/2, 0}, {3, 5/2 - (3 Sqrt[3])/2,
0}, {3, 1/2 - Sqrt[3]/2, 0}}]
Graphics3D[{{Thick, Red, xp3}, {Dashed, xs3}}, Axes -> True]
Normal@xp3

```

```

xp3vektor1 = -{1, 3/2 - Sqrt[3]/2, 1/2 + (3 Sqrt[3])/2} + {3,
  5/2 - (3 Sqrt[3])/2, 3/2 + (5 Sqrt[3])/2}
xp3vektor2 = +{3, 5/2 - (3 Sqrt[3])/2, 3/2 + (5 Sqrt[3])/2} -
{3, 1/2 - Sqrt[3]/2, 1/2 + Sqrt[3]/2}
N[VectorAngle[xp3vektor1, xp3vektor2]]
xs3vektor1 = -{1, 3/2 - Sqrt[3]/2, 0} + {3, 5/2 - (3 Sqrt[3])/
2, 0}
xs3vektor2 = +{3, 5/2 - (3 Sqrt[3])/2, 0} - {3, 1/2 - Sqrt[3]/
2, 0}
N[VectorAngle[xs3vektor1, xs3vektor2]]
RotationMatrix[Pi/2, {1, 0, 0}] // MatrixForm
xp4 := Rotate[p1, Pi/2, {1, 0, 0}]
xs4 = Line[{{1, -1, 0}, {3, -3, 0}, {3, -1, 0}}]
Graphics3D[{{Thick, Red, xp4}, {Dashed, xs4}}, Axes -> True]
Normal@xp4
xp4vektor1 = -{1, -1, 3} + {3, -3, 5}
xp4vektor2 = +{3, -3, 5} - {3, -1, 1}
N[VectorAngle[xp4vektor1, xp4vektor2]]
xs4vektor1 = -{1, -1, 0} + {3, -3, 0}
xs4vektor2 = +{3, -3, 0} - {3, -1, 0}
N[VectorAngle[xs4vektor1, xs4vektor2]]
RotationMatrix[2 Pi/3, {1, 0, 0}] // MatrixForm
xp5 := Rotate[p1, 2 Pi/3, {1, 0, 0}]
xs5 = Line[{{1, -(3/2) - Sqrt[3]/2, 0}, {3, -(5/2) - (3
Sqrt[3])/2, 0}, {3, -(1/2) - Sqrt[3]/2, 0}}]
Graphics3D[{{Thick, Red, xp5}, {Dashed, xs5}}, Axes -> True]
Normal@xp5
xp5vektor1 = -{1, -(3/2) - Sqrt[3]/2, -(1/2) + (3 Sqrt[3])/2}
+ {3, -(5/2) - (3 Sqrt[3])/2, -(3/2) + (5 Sqrt[3])/2}
xp5vektor2 = +{3, -(5/2) - (3 Sqrt[3])/2, -(3/2) + (5
Sqrt[3])/2} - {3, -(1/2) - Sqrt[3]/2, -(1/2) + Sqrt[3]/2}
N[VectorAngle[xp5vektor1, xp5vektor2]]
xs5vektor1 = -{1, -(3/2) - Sqrt[3]/2, 0} + {3, -(5/2) - (3
Sqrt[3])/2, 0}
xs5vektor2 = +{3, -(5/2) - (3 Sqrt[3])/2, 0} - {3, -(1/2) -
Sqrt[3]/2, 0}
N[VectorAngle[xs5vektor1, xs5vektor2]]

```

6.2 IZRIS ROGOVILE IN NJENIH PROJEKCIJ

```

x11 = Line[{{0, 3, 3}, {3, 3, 3}}]
x12 = Line[{{0, 6, 1}, {3, 6, 1}}]
x13 = Line[{{0, 3, 1}, {1, 3, 1}}]
x14 = Line[{{0, 2, 0}, {6, 2, 0}}]
y11 = Line[{{3, 0, 3}, {3, 3, 3}}]
y12 = Line[{{3, 0, 1}, {3, 6, 1}}]
y13 = Line[{{1, 0, 1}, {1, 3, 1}}]
y14 = Line[{{6, 0, 0}, {6, 2, 0}}]
z11 = Line[{{3, 3, 0}, {3, 3, 3}}]
z12 = Line[{{3, 6, 0}, {3, 6, 1}}]
z13 = Line[{{1, 3, 0}, {1, 3, 1}}]
z14 = Line[{{6, 2, 0}, {6, 2, 0}}]
ga1 = Arrow[Tube[{{3, 3, 3}, {3, 6, 1}}, 0.05]]
ga2 = Arrow[Tube[{{3, 3, 3}, {1, 3, 1}}, 0.05]]
ga3 = Arrow[Tube[{{3, 3, 3}, {6, 2, 0}}, 0.05]]
xa1 = Arrow[{{0, 3, 3}, {0, 6, 1}}]
xa2 = Arrow[{{0, 3, 3}, {0, 3, 1}}]
xa3 = Arrow[{{0, 3, 3}, {0, 2, 0}}]
ya1 = Arrow[{{3, 0, 3}, {3, 0, 1}}]
ya2 = Arrow[{{3, 0, 3}, {1, 0, 1}}]
ya3 = Arrow[{{3, 0, 3}, {6, 0, 0}}]
za1 = Arrow[{{3, 3, 0}, {3, 6, 0}}]
za2 = Arrow[{{3, 3, 0}, {1, 3, 0}}]
za3 = Arrow[{{3, 3, 0}, {6, 2, 0}}]
Graphics3D[{{Blue, ga1}, {Red, ga2}, {Green, ga3}}]
Graphics3D[{{Thick, {Blue, xa1}, {Red, xa2}, {Green, xa3},
{Blue, ya1}, {Red,
    ya2}, {Green, ya3}, {Blue, za1}, {Red, za2}, {Green,
za3}}}]
Graphics3D[{{Blue, ga1}, {Red, ga2}, {Green,
    ga3}, {Thick, {Blue, xa1}, {Red, xa2}, {Green, xa3}, {Blue,
ya1}, {Red,
    ya2}, {Green, ya3}, {Blue, za1}, {Red, za2}, {Green,
za3}}}]
Graphics3D[{{Blue, ga1}, {Red, ga2}, {Green,
    ga3}, {Thick, {Blue, xa1}, {Red, xa2}, {Green, xa3}, {Blue,
ya1}, {Red,
    ya2}, {Green, ya3}, {Blue, za1}, {Red, za2}, {Green,
za3}}, {Dashed,
    x11}, {Dashed, x12}, {Dashed, x13}, {Dashed, x14}, {Dashed,
y11}, {Dashed,
    y12}, {Dashed, y13}, {Dashed, y14}, {Dashed, z11}, {Dashed,
z12}, {Dashed,
    z13}, {Dashed, z14}}]

```

6.3 REKONSTRUKCIJA ENE STRANICE Z METODO SEŠTEVANJA VEKTORJEV

```

s = {3, 3, 3}
sx = {0, 3, 3}
sbx = {0, 2, 0}
sy = {3, 0, 3}
sby = {6, 0, 0}
vsx = sx - sbx
vsy = sy - sby
rez = vsx + vsy
temp = rez[[3]]
rez[[3]] = temp/2
Graphics3D[{Thick, {Dashed, Line[{{3, 3, 3}, {1, 3, 1}}]},
  Line[{{3, 3, 3}, {3, 6, 1}}], {Red, Dashed,
  Line[{{3, 3, 3}, {6, 2, 0}}]}, {Dashed, Line[{{0, 3, 3},
{0, 3, 1}}]},
  Line[{{0, 3, 3}, {0, 6, 1}}], {Red, Line[{{0, 3, 3}, {0, 2,
0}}]}, {Dashed,
  Line[{{3, 0, 3}, {1, 0, 1}}]},
  Line[{{3, 0, 3}, {3, 0, 1}}], {Red, Line[{{3, 0, 3}, {6, 0,
0}}]}, {Green,
  Line[{{s, (s - rez)}}]}, Axes -> True, AxesLabel -> {x, y,
z}]

```

6.4 REKONSTRUKCIJA ENE STRANICE Z METODO SEKANJA RAVNIN

```

matrixp = {{3, 3, 3}, {1, 3, 1}, {3, 3, 3}}
matrix2 = matrixp; matrix2[[All, 1]] = 0; matrix2[[1, 1]] = 3
matrix2x = matrix2; matrix2x[[All, 1]] = 1
matrix2y = matrix2; matrix2y[[All, 2]] = 1
matrix2z = matrix2; matrix2z[[All, 3]] = 1
matrix3 = matrixp; matrix3[[All, 2]] = 0; matrix3[[1, 2]] = 3
matrix3x = matrix3; matrix3x[[All, 1]] = 1
matrix3y = matrix3; matrix3y[[All, 2]] = 1
matrix3z = matrix3; matrix3z[[All, 3]] = 1
test2 = Det[matrix2x] x + Det[matrix2y] y + Det[matrix2z] z -
Det[matrix2]
test3 = Det[matrix3x] x + Det[matrix3y] y + Det[matrix3z] z -
Det[matrix3]
smer = Cross[{Det[matrix2x], Det[matrix2y], Det[matrix2z]},
{Det[matrix3x],
  Det[matrix3y], Det[matrix3z]}]
Graphics3D[{Thick, {Red, Line[{{3, 3, 3}, {1, 3, 1}]}],
{Green,
  Line[{{3, 3, 3}, {3, 6, 1}]}], {Blue, Line[{{3, 3, 3}, {6,
2, 0}]}], {Red,
  Line[{{0, 3, 3}, {0, 3, 1}]}], {Green,
  Line[{{0, 3, 3}, {0, 6, 1}]}], {Blue, Line[{{0, 3, 3}, {0,
2, 0}]}], {Red,
  Line[{{3, 0, 3}, {1, 0, 1}]}], {Green,
  Line[{{3, 0, 3}, {3, 0, 1}]}], {Blue,
  Line[{{3, 0, 3}, {6, 0, 0}]}], {Black, Dashed,
  Line[{{3, 3, 3}, {3, 3, 3} - smer/10]}], Thin,
  First[ContourPlot3D[{test2 == 0}, {x, 0, 7}, {y, 0, 7}, {z,
0, 7}]],
  First[ContourPlot3D[{test3 == 0}, {x, 0, 7}, {y, 0, 7}, {z,
0, 7}]]],
  Axes -> True, AxesLabel -> {x, y, z}]
(*smer premice, ki je rezultanta preseka dveh ravnin je
vektorski produkt \
normal obeh ravnin*)

```

6.5 PRIMERJAVA METODE SEŠTEVANJA VEKTORJEV IN METODE SEKANJA RAVNIN

```

s = {3, 3, 3}
a = {1, 3, 1}
sx = {0, 3, 3}
sy = {3, 0, 3}
ax = {0, 3, -1}
ay = {1, 0, 1}
ravnina1 = {s, ax, sx}
ravnina1x = ravnina1; ravnina1x[[All, 1]] = 1
ravnina1y = ravnina1; ravnina1y[[All, 2]] = 1
ravnina1z = ravnina1; ravnina1z[[All, 3]] = 1
ravnina2 = {s, ay, sy}
ravnina2x = ravnina2; ravnina2x[[All, 1]] = 1
ravnina2y = ravnina2; ravnina2y[[All, 2]] = 1
ravnina2z = ravnina2; ravnina2z[[All, 3]] = 1
vsx = sx - ax
vsy = sy - ay
rez = vsx + vsy
temp = rez[[3]]
rez[[3]] = temp/2
smer = Cross[{Det[ravnina1x], Det[ravnina1y],
  Det[ravnina1z]}, {Det[ravnina2x], Det[ravnina2y],
  Det[ravnina2z]}]
Graphics3D[{Thick, {Dashed, Blue, Line[{{3, 3, 3}, {1, 3,
1}]}]},
  Line[{{3, 3, 3}, {3, 6, 1}}], {Red, Dashed,
  Line[{{3, 3, 3}, {6, 2, 0}}]}, {Dashed, Blue,
  Line[{{0, 3, 3}, {0, 3, -1}}]},
  Line[{{0, 3, 3}, {0, 6, 1}}], {Red, Line[{{0, 3, 3}, {0, 2,
0}}]}, {Dashed,
  Blue, Line[{{3, 0, 3}, {1, 0, 1}}]},
  Line[{{3, 0, 3}, {3, 0, 1}}], {Red, Line[{{3, 0, 3}, {6, 0,
0}}]}, {Green,
  Line[{{s, (s - rez)}}], {Orange, Line[{{s, (s - smer/30)}}]},
Axes -> True,
  AxesLabel -> {x, y, z}]

```

6.6 REKONSTRUKCIJA RAZCEPIŠČA IN IZRAČUN KOTOV

```

s = {33, 33, 33}
a = {11, 32, 13}
b = {34, 66, 10}
c = {65, 26, 3}
zamikX = {0, 3, 4}
zamikY = {-2, 0, -1}
sx = s; sx[[1]] = 0; sx = sx + zamikX
sy = s; sy[[2]] = 0; sy = sy + zamikY
ax = a; ax[[1]] = 0; ax = ax + zamikX
vektorSAX = sx - ax
ax = sx - 1/2 vektorSAX
ay = a; ay[[2]] = 0; ay = ay + zamikY
bx = b; bx[[1]] = 0; bx = bx + zamikX
by = b; by[[2]] = 0; by = by + zamikY
cx = c; cx[[1]] = 0; cx = cx + zamikX
cy = c; cy[[2]] = 0; cy = cy + zamikY
(*---rekonstrukcija točke S---*)
normalaX = {1, 0, 0}
normalaY = {0, 1, 0}
smerpremika = -Cross[normalaX, normalaY]/
  Sqrt[Dot[Cross[normalaX, normalaY], Cross[normalaX,
normalaY]]]
pomožnaMatrika = {sx - sy, normalaX, normalaY} // MatrixForm
razdalja =
  Det[{sx - sy, normalaX, normalaY}]/
  Sqrt[Power[
    Det[{{normalaX[[1]], normalaX[[2]]}, {normalaY[[1]],
normalaY[[2]]}}, 2] +
    Power[Det[{{normalaX[[2]], normalaX[[3]]}, {normalaY[[2]],
normalaY[[3]]}}, 2] +
    Power[Det[{{normalaX[[3]], normalaX[[1]]}, {normalaY[[3]],
normalaY[[2]]}}, 2]]]
(*prvo premico premaknemo v smeri premika za izračunano
razdaljo*)
premica1 = sx + normalaX*spr1
premica2 = sy + normalaY*spr2
premica1p = smerpremika*razdalja + premica1
Solve[{premica2[[1]] == premica1p[[1]], premica2[[2]] ==
premica1p[[2]], premica2[[3]] == premica1p[[3]]}, {spr1,
spr2}]
espr1 = spr1 /. %
espr2 = spr2 /. %%
espr1 = First[espr1]
espr2 = First[espr2]
premica1 = sx + normalaX*espr1
premica2 = sy + normalaY*espr2
(*dobimo točki t1 ter t2 aritmetična sredina pa je rS*)

```

```

rS = (premica1 + premica2)/2
pomožnarSx = (normalaX[[1]] rS[[1]] + normalaX[[2]] rS[[2]] +
  normalaX[[3]] rS[[3]] - normalaX[[1]] sx[[1]] -
normalaX[[2]] sx[[2]] -
  normalaX[[3]] sx[[3]])/(normalaX[[1]] normalaX[[1]] +
  normalaX[[2]] normalaX[[2]] + normalaX[[3]] normalaX[[3]])
rSx = {rS[[1]] - normalaX[[1]] pomožnarSx, rS[[2]] -
normalaX[[2]] pomožnarSx, rS[[3]] - normalaX[[3]] pomožnarSx}
pomožnarSy = (normalaY[[1]] rS[[1]] + normalaY[[2]] rS[[2]] +
  normalaY[[3]] rS[[3]] - normalaY[[1]] sy[[1]] -
normalaY[[2]] sy[[2]] -
  normalaY[[3]] sy[[3]])/(normalaY[[1]] normalaY[[1]] +
  normalaY[[2]] normalaY[[2]] + normalaY[[3]] normalaY[[3]])
rSy = {rS[[1]] - normalaY[[1]] pomožnarSy, rS[[2]] -
normalaY[[2]] pomožnarSy,
  rS[[3]] - normalaY[[3]] pomožnarSy}
korekcijskiVektorX = rSx - sx
korekcijskiVektorY = rSy - sy
(*s korekcijskima vektorjema sedaj popravimo vse točke*)
rax = ax + korekcijskiVektorX
rbx = bx + korekcijskiVektorX
rcx = cx + korekcijskiVektorX
ray = ay + korekcijskiVektorY
rby = by + korekcijskiVektorY
rcy = cy + korekcijskiVektorY
(*---konec rekonstrukcije---*)
ravninal = {rS, rax, rSx}
ravninalx = ravninal; ravninalx[[All, 1]] = 1
ravninaly = ravninal; ravninaly[[All, 2]] = 1
ravninalz = ravninal; ravninalz[[All, 3]] = 1
ravnina2 = {rS, ray, rSy}
ravnina2x = ravnina2; ravnina2x[[All, 1]] = 1
ravnina2y = ravnina2; ravnina2y[[All, 2]] = 1
ravnina2z = ravnina2; ravnina2z[[All, 3]] = 1
ravninab1 = {rS, rbx, rSx}
ravninab1x = ravninab1; ravninab1x[[All, 1]] = 1
ravninab1y = ravninab1; ravninab1y[[All, 2]] = 1
ravninab1z = ravninab1; ravninab1z[[All, 3]] = 1
ravninab2 = {rS, rby, rSy}
ravninab2x = ravninab2; ravninab2x[[All, 1]] = 1
ravninab2y = ravninab2; ravninab2y[[All, 2]] = 1
ravninab2z = ravninab2; ravninab2z[[All, 3]] = 1
ravninac1 = {rS, rcx, rSx}
ravninac1x = ravninac1; ravninac1x[[All, 1]] = 1
ravninac1y = ravninac1; ravninac1y[[All, 2]] = 1
ravninac1z = ravninac1; ravninac1z[[All, 3]] = 1
ravninac2 = {rS, rcy, rSy}
ravninac2x = ravninac2; ravninac2x[[All, 1]] = 1
ravninac2y = ravninac2; ravninac2y[[All, 2]] = 1
ravninac2z = ravninac2; ravninac2z[[All, 3]] = 1

```

```

smer = Cross[{Det[ravnina1x], Det[ravnina1y],
  Det[ravnina1z]}, {Det[ravnina2x], Det[ravnina2y],
  Det[ravnina2z]}]
normalaRavnineA = rSx - rax
pomožnaSprA = (normalaRavnineA[[1]] rS[[1]] +
normalaRavnineA[[2]] rS[[2]] +
  normalaRavnineA[[3]] rS[[3]] - normalaRavnineA[[1]]
rax[[1]] - normalaRavnineA[[2]] rax[[2]] -
  normalaRavnineA[[3]] rax[[3]]) / (normalaRavnineA[[1]]
smer[[1]] + normalaRavnineA[[2]] smer[[2]] +
normalaRavnineA[[3]] smer[[3]])
aRekonstruirana = {rS[[1]] - smer[[1]] pomožnaSprA,
  rS[[2]] - smer[[2]] pomožnaSprA, rS[[3]] - smer[[3]]
pomožnaSprA}
smerb = Cross[{Det[ravninab1x], Det[ravninably],
  Det[ravninab1z]}, {Det[ravninab2x], Det[ravninab2y],
  Det[ravninab2z]}]
normalaRavnineB = rSx - rbx
pomožnaSprB = (normalaRavnineB[[1]] rS[[1]] +
normalaRavnineB[[2]] rS[[2]] + normalaRavnineB[[3]] rS[[3]] -
normalaRavnineB[[1]] rbx[[1]] - normalaRavnineB[[2]] rbx[[2]]
- normalaRavnineB[[3]] rbx[[3]]) / (normalaRavnineB[[1]]
smerb[[1]] + normalaRavnineB[[2]] smerb[[2]] +
normalaRavnineB[[3]] smerb[[3]])
bRekonstruirana = {rS[[1]] - smerb[[1]] pomožnaSprB, rS[[2]] -
smerb[[2]] pomožnaSprB, rS[[3]] - smerb[[3]] pomožnaSprB}
smerc = Cross[{Det[ravninac1x], Det[ravninac1y],
  Det[ravninac1z]}, {Det[ravninac2x], Det[ravninac2y],
  Det[ravninac2z]}]
normalaRavnineC = rSx - rcx
pomožnaSprC = (normalaRavnineC[[1]] rS[[1]] +
normalaRavnineC[[2]] rS[[2]] +
  normalaRavnineC[[3]] rS[[3]] - normalaRavnineC[[1]]
rcx[[1]] - normalaRavnineC[[2]] rcx[[2]] -
  normalaRavnineC[[3]] rcx[[3]]) / (normalaRavnineC[[1]]
smerc[[1]] + normalaRavnineC[[2]] smerc[[2]] +
normalaRavnineC[[3]] smerc[[3]])
cRekonstruirana = {rS[[1]] - smerc[[1]] pomožnaSprC,
  rS[[2]] - smerc[[2]] pomožnaSprC, rS[[3]] - smerc[[3]]
pomožnaSprC}
Graphics3D[{{Thick, {Blue, {Tube[{s, a}, 0.5]}, Tube[{s, b},
0.5],
  Tube[{s, c}, 0.5]}, Line[{sx, ax}], Line[{sx, bx}],
Line[{sx, cx}],
  Line[{sy, ay}], Line[{sy, by}],
Line[{sy, cy}], {Orange, {Line[{rS, aRekonstruirana}],
  Line[{rS, bRekonstruirana}], Line[{rS,
cRekonstruirana}}]}}, {Thick, Dashed,
  Red, {Line[{rSx, rax}], Line[{rSx, rbx}], Line[{rSx, rcx}],
Line[{rSy, ray}], Line[{rSy, rby}], Line[{rSy, rcy}]}}},

```

```
Axes -> True, AxesLabel -> {x, y, z}
(*kot med vektorjema SA in SB ter kot med rSrA ter rSrB*)
N[VectorAngle[s - a, s - b]]
N[VectorAngle[smer, smerb]]
(*kot med SB in SC ter rSrB in rSrC*)
N[VectorAngle[s - b, s - c]]
N[VectorAngle[smerb, smerc]]
```

6.7 REKONSTRUKCIJA RAZCEPIŠČA IZ SLIK (PRIMER SLIKA 22 IN 23)

```

alpha1 = -35*(180/Pi)
beta1 = 26 *(180/Pi)
sxNerotirana = {129, 107, 0}
axNerotirana = {34, 128, 0}
bxNerotirana = {227, 94, 0}
cxNerotirana = {196, 173, 0}
alpha2 = 47*(180/Pi)
beta2 = 29*(180/Pi)
syNerotirana = {201, 204, 0}
ayNerotirana = {136, 289, 0}
byNerotirana = {320, 177, 0}
cyNerotirana = {234, 104, 0}
rotacijaAlphaX1 = RotationTransform[alpha1, {1, 0, 0}] // N
rotacijaBetaY1 = RotationTransform[beta1, {0, 1, 0}] // N
rotacijaAlphaX2 = RotationTransform[alpha2, {1, 0, 0}] // N
rotacijaBetaY2 = RotationTransform[beta2, {0, 1, 0}] // N
sxDelnoRotirana = rotacijaAlphaX1[sxNerotirana]
sx = rotacijaBetaY1[sxDelnoRotirana]
axDelnoRotirana = rotacijaAlphaX1[axNerotirana]
ax = rotacijaBetaY1[axDelnoRotirana]
bxDelnoRotirana = rotacijaAlphaX1[bxNerotirana]
bx = rotacijaBetaY1[bxDelnoRotirana]
cxDelnoRotirana = rotacijaAlphaX1[cxNerotirana]
cx = rotacijaBetaY1[cxDelnoRotirana]
syDelnoRotirana = rotacijaAlphaX2[syNerotirana]
sy = rotacijaBetaY2[syDelnoRotirana]
ayDelnoRotirana = rotacijaAlphaX2[ayNerotirana]
ay = rotacijaBetaY2[ayDelnoRotirana]
byDelnoRotirana = rotacijaAlphaX2[byNerotirana]
by = rotacijaBetaY2[byDelnoRotirana]
cyDelnoRotirana = rotacijaAlphaX2[cyNerotirana]
cy = rotacijaBetaY2[cyDelnoRotirana]
(*---rekonstrukcija točke S---*)
Clear[vax, vbx, vay, vby]
vax = Normalize[sx - ax]
vbx = Normalize[sx - bx]
vay = Normalize[sy - ay]
vby = Normalize[sy - by]
normalaX = Cross[vax, vbx]
normalaY = Cross[vay, vby]
(*koda je od te točke naprej enaka kodi v poglavju 5.6*)

```

7 RAZPRAVA

Rekonstrukcija razcepišča srčnih žil, je netrivialen problem, saj se srce med snemanjem premika, same žile se zaradi elastičnosti deformirajo, slike koronarnih žil posnete pod različnimi kotih imajo ponavadi majhen zamik, na samem razcepišču lahko z gotovostjo označimo le eno točko (točko središča razcepišča) ...

Ugotovil sem, da je kljub vsem tem preprekam mogoče rekonstruirati tridimenzionalni model razcepišča z algoritmi in postopki, ki smo jih razvili v tej diplomski nalogi. Meritve kotov med žilami v razcepišču s pomočjo tridimenzionalnega modela upoštevajo popačitve projiciranih dolžin žil in velikosti kotov in se v nasprotju s trenutno najpogosteje uporabljenimi metodami ne zanašajo na izkušnje kardiologa. Metode, ki uporabljajo za meritev zgolj eno projekcijo za vsak izmerjen kot, v praksi eno projekcijo za merjenje kota med distalno glavno arterijo in stransko vejo ter eno projekcijo za kot med stransko vejo in proksimalno glavno arterijo, delujejo natančno le, kadar je pogled na dani žili ortogonalen oziroma sta žili v ravnini, ki je z detektorjem vzporedna. Določitev pozicije detektorja, ki bi tak pogled na razcepišče omogočala, pa je prepuščena izkušnosti kardiologa in smernicam izdanim na podlagi statističnih opažanj iz večih meritev, le te ne upoštevajo morebitne anatomsko nepravilne lege srca ali majhnih odstopanj med posameznimi pacienti. Za natančno ocenitev kotov v razcepišču je torej potrebna tridimenzionalna rekonstrukcija. Postopek, ki smo ga razvili v ta namen tako zmanjša vpliv človeške napake in omogoča lažjo primerjavo med različnimi meritvami (npr.: primerjavo med izmerjenimi koti razcepišča pred in po vstavitvi žilne opornice).

8 LITERATURA

- [1] Issam D. Moussa, Antonio Colombo, *Tips and Tricks in Interventional Therapy of Coronary Bifurcation Lesions*, Informa Healthcare: London, 2010, pogl. 2.
- [2] I. N. Bronstein, K. A. Semendjajew, G. Musiol, H. Mühlig, *Matematični Priručnik*, Tehniška založba Slovenije, 1997, str. 183-187.
- [3] Weisstein, Eric W. "Skew Lines.", MathWorld--A Wolfram Web Resource. Dostopno na: <http://mathworld.wolfram.com/SkewLines.html> (24.12.2011)
- [4] Weisstein, Eric W. "Plane-Plane Intersection.", MathWorld--A Wolfram Web Resource. Dostopno na: <http://mathworld.wolfram.com/Plane-PlaneIntersection.html> (24.11.2011)
- [5] Weisstein, Eric W. "Line-Plane Intersection.", MathWorld--A Wolfram Web Resource. Dostopno na: <http://mathworld.wolfram.com/Line-PlaneIntersection.html> (24.11.2011)
- [6] Weisstein, Eric W. "Plane.", MathWorld--A Wolfram Web Resource. Dostopno na: <http://mathworld.wolfram.com/Plane.html> (24.11.2011)
- [7] Weisstein, Eric W. "Vector.", MathWorld--A Wolfram Web Resource. Dostopno na: <http://mathworld.wolfram.com/Vector.html> (24.11.2011)
- [8] Ed Pegg Jr. "From Vector to Plane", Wolfram Demonstrations Project. Dostopno na: <http://demonstrations.wolfram.com/FromVectorToPlane/> (2.1.2012)
- [9] Foley, van Dam, Feiner, Hugher, *Computer Graphics Principles and Practice*, Addison-Wesley Publishing Company, Inc., 1997, pogl. 5.