



**MODELISASI KOMPONEN SAPU
MELALUI PERMUKAAN PUTAR DAN TRANSFORMASI
BENDA DASAR GEOMETRI BIDANG**

SKRIPSI

Oleh

**Diana Nurfarida
NIM. 121810101033**

**JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS JEMBER
2016**



**MODELISASI KOMPONEN SAPU
MELALUI PERMUKAAN PUTAR DAN TRANSFORMASI
BENDA DASAR GEOMETRI BIDANG**

SKRIPSI

diajukan guna melengkapi tugas akhir dan memenuhi salah satu syarat
untuk menyelesaikan Program Studi Matematika (S1)
dan mencapai gelar Sarjana Sains

Oleh

**Diana Nurfarida
NIM. 121810101033**

**JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS JEMBER
2016**

PERSEMBAHAN

Alhamdulillah, dengan puji syukur kehadirat Allah SWT, skripsi ini saya persembahkan untuk:

1. Ibunda Nur Wachidah dan Ayahanda Ahmad Sholeh tercinta, yang telah mendoakan dan memberi kasih sayang serta pengorbanan untuk putri tercintanya;
2. Nenek Kati, Kakek Marli, dan Adik M. Irfan Jauhari tersayang yang telah banyak memberikan semangat dalam penyelesaian skripsi ini;
3. guru-guruku sejak taman kanak-kanak sampai perguruan tinggi, yang telah memberikan ilmu dan membimbing dengan penuh kesabaran;
4. Almamater Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember, MA Negeri 3 Jember, SMP Negeri 1 Umbulsari, MI Nurul Huda Gunungsari, dan TK Dewi Masyithah Banjarejo.

MOTTO

Katakanlah, “Kalau sekiranya lautan menjadi tinta untuk (menulis) kalimat-kalimat Tuhan-ku, sungguh habislah lautan itu sebelum habis (ditulis) kalimat-kalimat Tuhan-ku, meskipun Kami datangkan tambahan sebanyak itu (pula)“.

(QS. Al-Kahf: 109)¹

“Barang siapa yang keluar untuk mencari ilmu maka dia berada di jalan Allah hingga ia pulang”

(HR. Tirmidzi)²

¹ Departemen Agama Republik Indonesia. 2004. Al Qur'an dan Terjemahannya: Al-Jumanatul Ali. Bandung: CV Penerbit J-ART

² Setiyawan, Risky. 2015. Hadits tentang Ilmu [serial online].
<https://riskyds.wordpress.com/about/hadits-tentang-ilmu/>. [24 Juni 2016]

PERNYATAAN

Saya yang bertanda tangan di bawah ini :

nama : Diana Nurfarida

NIM : 121810101033

menyatakan dengan sesungguhnya bahwa skripsi yang berjudul “Modelisasi Komponen Sapu Melalui Permukaan Putar dan Transformasi Benda Dasar Geometri Bidang” adalah benar-benar hasil karya sendiri, kecuali kutipan yang sudah saya sebutkan sumbernya, belum pernah diajukan pada institusi mana pun, dan bukan karya jiplakan. Saya bertanggung jawab atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

Demikian pernyataan ini saya buat dengan sebenarnya, tanpa ada tekanan dan paksaan dari pihak mana pun serta bersedia mendapat sanksi akademik jika ternyata di kemudian hari pernyataan ini tidak benar.

Jember, 24 Juni 2016
Yang menyatakan,

Diana Nurfarida
NIM 121810101033

SKRIPSI

**MODELISASI KOMPONEN SAPU
MELALUI PERMUKAAN PUTAR DAN TRANSFORMASI
BENDA DASAR GEOMETRI BIDANG**

Oleh

Diana Nurfarida
NIM. 121810101033

Pembimbing

Dosen Pembimbing Utama : Prof. Drs. Kusno, DEA., Ph.D.

Dosen Pembimbing Anggota : Ika Hesti Agustin, S.Si., M.Si.

PENGESAHAN

Skripsi berjudul "Modelisasi Komponen Sapu Melalui Permukaan Putar dan Transformasi Benda Dasar Geometri Bidang" telah diuji dan disahkan pada:

hari :

tanggal :

tempat : Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

Tim Penguji :

Dosen Pembimbing Utama,

Dosen Pembimbing Anggota,

Prof. Drs. Kusno, DEA., Ph.D.
NIP 19610108 198602 1 001

Ika Hesti Agustin, S.Si., M.Si.
NIP 19840801 200801 2 006

Penguji I,

Penguji II,

Drs. Rusli Hidayat, M.Sc.
NIP 19661012 199303 1 001

Dr. Moh. Fatekurohman, S.Si., M.Si.
NIP 19690606 199803 1 001

Mengesahkan
Dekan,

Drs. Sujito, Ph.D.
NIP 19610204 198711 1 001

RINGKASAN

Modelisasi Komponen Sapu Melalui Permukaan Putar dan Transformasi Benda Dasar Geometri Bidang; Diana Nurfarida; 2016; 58 Halaman; Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

Sapu merupakan salah satu peralatan kebersihan yang digunakan untuk aktifitas rumah tangga sehari-hari. Ada berbagai macam jenis sapu sesuai bentuk, ukuran, dan fungsinya antara lain sapu ijuk lantai, sapu mobil, sapu langit-langit, dan lain sebagainya. Umumnya dari keseluruhan model yang telah dibuat, bentuk tongkat, lakop, dan ijuk sapu memiliki kekurangan. Hal ini dapat dilihat dari tongkat sapu yang memiliki bentuk monoton dan hanya termodelisasi dari tabung sehingga mudah patah. Begitu pula dengan bagian lakop yang belum memiliki variasi bentuk, biasanya hanya berupa lakop lurus dan lakop setengah lingkaran atau lakop bulan. Penulisan skripsi ini dimaksudkan untuk memodelisasi bentuk komponen sapu melalui transformasi dan permukaan putar dari benda geometri bidang sehingga menghasilkan sapu yang variatif bentuknya.

Dalam penelitian modelisasi sapu ini dibagi menjadi beberapa tahapan. Tahapan pertama adalah memodelisasikan benda dasar geometri untuk membangun bagian-bagian komponen sapu yaitu bagian tongkat, lakop dan ijuk. Untuk bagian tongkat dengan cara menetapkan data penelitian berupa segmen garis, kemudian membangun permukaan tongkat dengan kurva dan menginterpolasikannya. Untuk bagian lakop dan ijuk dengan cara menetapkan data persegi panjang, kemudian mentranslasi dan mendilatasi lalu menginterpolasikannya secara linier. Tahapan kedua adalah penggabungan ketiga bagian komponen sapu hasil modelisasi melalui satu sumbu pemodelan. Tahapan ketiga menyusun program komputer menggunakan Maple 18.

Hasil penelitian ini didapatkan dua prosedur untuk memodelisasi sapu yaitu mendesain beragam bentuk sapu yang terdiri dari 3 komponen yaitu tongkat, lakop, dan ijuk, prosedurnya sebagai berikut. Pertama, membangun bentuk permukaan tongkat sapu dari segmen vertikal dan persegi di kedua ujungnya melalui grid

persegi dan potongan tegak dari segmen vertikal, kemudian menginterpolasi pasangan kurva secara berurutan. Kedua, mengkonstruksi bagian lakop melalui translasi dan dilatasi persegi panjang yang dilengkungkan dengan kurva Bezier, kemudian menginterpolasikan persegi panjang sebelum dan sesudah translasi. Ketiga, mengkonstruksi ijuk sapu dari persegi panjang dasar lakop, kemudian menginterpolasikan titik-titik didalam grid persegi panjang. Selanjutnya merangkai sapu ijuk secara utuh diperlukan unsur kesimetrisan melalui satu sumbu pemodelan, prosedurnya sebagai berikut. Pertama, membagi sumbu menjadi beberapa segmen. Kedua, mengisi setiap bagian segmen tersebut dengan bentuk dasar komponen sapu sehingga menghasilkan model sapu yang bervariasi.

PRAKATA

Puji syukur kehadiran Allah SWT, yang telah melimpahkan rahmat, dan hidayah-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul “Modelisasi Komponen Sapu Melalui Permukaan Putar dan Transformasi Benda Dasar Geometri Bidang”. Skripsi ini disusun untuk memenuhi salah satu syarat untuk menyelesaikan pendidikan strata satu (S1) pada Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

Penyusunan skripsi ini tidak lepas dari bantuan berbagai pihak. Oleh karena itu, penulis menyampaikan terima kasih kepada:

1. Prof. Drs. Kusno, DEA., Ph.D. selaku Dosen Pembimbing Utama dan Ika Hesti Agustin, S.Si., M.Si. selaku Dosen Pembimbing Anggota yang telah meluangkan waktu, pikiran, dan perhatian dalam penulisan skripsi ini;
2. Drs. Rusli Hidayat, M.Sc. dan Dr. Moh. Fatekurohman, S.Si., M.Si. selaku Dosen Penguji yang telah memberikan kritikan dan saran demi kesempurnaan skripsi ini;
3. seluruh staf pengajar Jurusan Matematika Fakultas MIPA Universitas Jember yang telah memberikan ilmu serta bimbingannya sehingga penulis dapat menyelesaikan skripsi ini;
4. keluarga tercinta Ibu Nur Wachidah, Ayah Ahmad Sholeh, Nenek Kati, Kakek Marli, dan Adik M. Irfan Jauhari yang telah memberi dukungan sepenuhnya terhadap saya semasa hidup baik dari segi spiritual maupun material;
5. sahabat terbaik (Silvia, Firda, Ummah, dan Tri) yang telah memberikan dorongan dan semangat untuk menyelesaikan tugas akhir ini;
6. teman-teman BATHIC’S 12, kakak dan adik angkatan, serta teman-teman yang lainnya, terima kasih atas kebersamaan selama waktu kuliah dan telah memberikan semangat serta motivasi;
7. sahabat kostan Kalimantan 72 (Anggrek dan Iyuk) yang telah memberi dukungan dan kebahagiaan saat kita bersama;
8. semua pihak yang tidak dapat disebutkan satu per satu.

Penulis juga menerima segala kritik dan saran dari semua pihak demi kesempurnaan skripsi ini. Akhirnya penulis berharap, semoga skripsi ini dapat bermanfaat.

Jember, Juni 2016

Penulis



DAFTAR ISI

	Halaman
HALAMAN JUDUL	ii
HALAMAN PERSEMBAHAN	iii
HALAMAN MOTTO	iv
HALAMAN PERNYATAAN	v
HALAMAN PEMBIMBINGAN	vi
HALAMAN PENGESAHAN	vii
RINGKASAN	viii
PRAKATA	x
DAFTAR ISI	xii
DAFTAR GAMBAR	xiv
BAB 1. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Tujuan Penelitian	6
1.4 Manfaat Penelitian	6
BAB 2. TINJAUAN PUSTAKA	7
2.1 Penyajian Segmen Garis	7
2.1.1 Segmen Garis di Bidang	7
2.1.2 Segmen Garis di Ruang	8
2.2 Penyajian Benda-benda Geometri di Bidang	9
2.2.1 Lingkaran	9
2.2.2 Elips	10
2.2.3 Persegi panjang	11
2.3 Transformasi Bidang di R^3	12
2.3.1 Rotasi (Perputaran)	12
2.3.2 Dilatasi (Penskalaan)	13
2.3.3 Translasi (Pergeseran).....	14
2.4 Interpolasi diantara Segmen Garis dan Kurva di Ruang	15

2.5 Kurva Hermit Kuadratik	16
2.6 Kurva Heliks.....	17
2.5 Kurva Bezier.....	17
2.6 Konstruksi Objek Dasar pada Program Maple 18.....	18
BAB 3. METODE PENELITIAN	23
BAB 4. HASIL DAN PEMBAHASAN	26
4.1 Desain Tongkat Sapu	26
4.1.1 Pola Lengkung Vertikal	26
4.1.2 Pola Potongan Tegak	32
4.1.3 Pola Puntiran.....	39
4.2 Desain Lakop dan Ijuk Sapu	40
4.2.1 Pola Dilatasi dan Translasi	40
4.2.2 Desain Ijuk.....	45
4.3 Penggabungan Komponen Penyusun Sapu	48
4.4 Pembahasan.....	53
BAB 5. PENUTUP	58
5.1 Kesimpulan	58
5.2 Saran	58
DAFTAR PUSTAKA	60

DAFTAR GAMBAR

	Halaman
1.1 Komponen sapu ijuk	2
1.2 Langkah-langkah modelisasi tongkat sapu	3
1.3 Modelisasi tongkat sapu bentuk potongan tegak dan puntiran	4
1.4 Modelisasi lakop sapu bentuk persegi panjang.....	5
1.5 Modelisasi ijuk sapu.....	6
2.1 Penyajian segmen garis di bidang.....	8
2.2 Penyajian segmen garis di ruang.....	8
2.3 Penyajian lingkaran.....	9
2.4 Penyajian keratan lingkaran	10
2.5 Penyajian elips	11
2.6 Penyajian persegi panjang.....	12
2.7 Ilustrasi rotasi pada sistem koordinat tangan kiri.....	13
2.8 Dilatasi dengan $k > 1$	14
2.9 Contoh kasus khusus interpolasi linier dan segmen garis.....	15
2.10 Interpolasi linier pada kurva.....	16
2.11 Kurva Hermit kuadratik	16
2.12 Kurva Heliks	17
2.13 Kurva Bezier	18
2.14 Penyajian segmen garis di bidang.....	18
2.15 Penyajian segmen garis di ruang.....	19
2.16 Penyajian lingkaran.....	19
2.17 Penyajian potongan lingkaran.....	20
2.18 Penyajian persegi panjang.....	20
2.19 Penyajian interpolasi	21
2.20 Penyajian kurva Hermit.....	21
2.21 Penyajian kurva Bezier	22
3.1 Skema metode penelitian	25
4.1 Langkah-langkah membangun pola lengkung vertikal.....	29

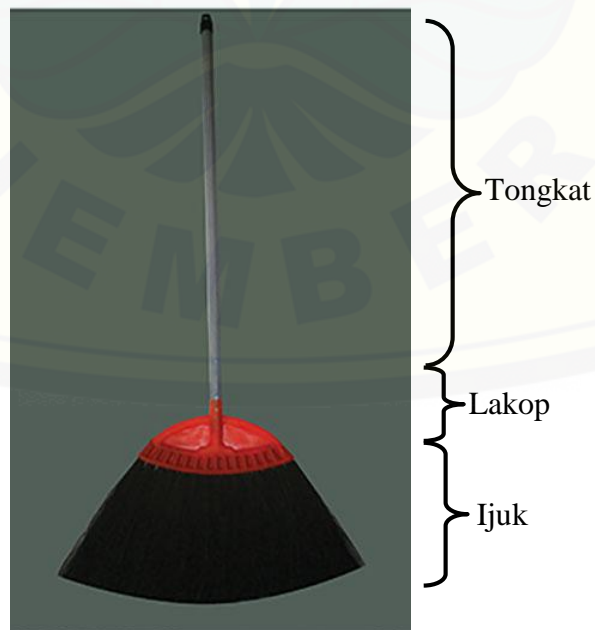
4.2	Variasi interpolasi dengan 2 kurva, 3 kurva, dan 4 kurva.....	30
4.3	Langkah-langkah interpolasi kurva ketinggian berbeda	31
4.4	Contoh visual permukaan hasil interpolasi linier pasangan kurva berketinggian berbeda	31
4.5	Langkah-langkah membuat pola potongan tegak	32
4.6	Langkah-langkah mengkonstruksi permukaan cekung-cembung.....	34
4.7	Variasi bentuk pola potongan tegak kasus cekung-cembung	35
4.8	Variasi kurva dari potongan lingkaran dan elips.....	37
4.9	Langkah-langkah mengkonstruksi tongkat kasus pola kepingan.....	38
4.10	Variasi pola kepingan kurva potongan elips dan lingkaran	38
4.11	Langkah-langkah interpolasi dengan sudut puntiran $\theta = 45^\circ$ berlawanan arah jarum jam	39
4.12	Visualisasi puntiran dengan sudut $\theta = 45^\circ$ berlawanan arah jarum jam	40
4.13	Langkah-langkah pola translasi dan dilatasi	42
4.14	Modelisasi persegi panjang menjadi lengkung menggunakan kurva Bezier	44
4.15	Langkah-langkah membuat lakop kasus pola lengkung	45
4.16	Visualisasi program variasi lakop pola lengkung dengan $t=10$	45
4.17	Langkah-langkah mengkonstruksi ijuk	47
4.18	Visualisasi interpolasi ijuk menggunakan Maple 18	48
4.19	Sumbu tegak penggabungan	51
4.20	Contoh penggabungan menggunakan satu sumbu penggabungan dengan $t_1 = 50cm, t_2 = 10cm, \text{ dan } t_3 = 10cm$	52
4.21	Contoh rangkaian sapu dengan satu sumbu pemodelan.....	53
4.22	Variasi bentuk tongkat sapu	54
4.23	Variasi bentuk lakop sapu	54
4.24	Variasi bentuk ijuk sapu.....	55
4.25	Penskalaan pada tiap tongkat pola potongan tegak.....	55
4.26	Contoh beberapa variasi penggabungan komponen sapu	57

BAB 1. PENDAHULUAN

1.1 Latar Belakang

Sapu merupakan salah satu peralatan kebersihan yang digunakan untuk aktifitas rumah tangga sehari-hari. Tidak hanya di rumah, sapu juga digunakan di perkantoran dan sekolah-sekolah. Sapu dapat digunakan untuk membersihkan bagian-bagian rumah maupun bagian dalam mobil sampai ke sudut yang paling sulit dijangkau. Seiring dengan kemajuan teknologi telah diciptakan beberapa alat pembersih secara elektrik berupa *vaccum cleaner*, *shampooing machine*, dan *floor machine*, tetapi sapu tetap banyak diminati oleh ibu rumah tangga. Hal ini dikarenakan selain harganya relatif murah, pemakaian dengan sapu lebih *simple* dan praktis, dan tidak memakan tempat dalam penyimpanannya.

Ada berbagai macam jenis sapu sesuai bentuk, ukuran, dan fungsinya antara lain sapu ijuk lantai, sapu mobil, sapu langit-langit, dan lain sebagainya. Sapu terdiri atas 3 bagian yaitu tongkat, lakop (bagian dasar untuk memasang ijuk), dan ijuk.



(a)Bagian-bagian sapu



(b)Lakop dan ijuk



(c)Tongkat

Sumber: www.bukalapak.com, www.tokopedia.com, dan www.franksupply.com

Gambar 1.1 Komponen sapu ijuk

Umumnya dari keseluruhan model yang ada, bentuk tongkat, lakop, dan ijuk sapu memiliki kekurangan. Hal ini dapat dilihat dari tongkat sapu yang memiliki bentuk monoton dan hanya termodelisasi dari tabung sehingga mudah patah. Begitu pula dengan bagian lakop yang belum memiliki variasi bentuk, biasanya hanya berupa lakop lurus dan lakop setengah lingkaran atau lakop bulan. Untuk itu modelisasi sapu ijuk bagian tongkat dan lakop ini diperlukan untuk menambah keindahan tanpa mengurangi nilai fungsinya.

Berdasarkan beberapa pemodelan yang telah dilakukan, terkait dengan bentuk bagian-bagian dari sapu, Dara (2015) membahas tentang modelisasi bros menggunakan lingkaran dan persegi. Namun model yang dikembangkan Dara hanya terbatas untuk transformasi geometri di bidang belum dikembangkan konstruksi transformasi di ruang. Kekurangan lain dari pemodelan Dara yaitu bahan dasar yang digunakan terbatas. Selanjutnya Dani (2012) mendesain vas bunga yang terdiri dari bagian atas vas, bagian utama, dan bagian alas. Namun model yang dikembangkan Dani untuk bagian alas vas bunga memiliki kelemahan yaitu hanya berupa poligon segienam beraturan, sehingga untuk pemodelan sapu dapat dikembangkan dengan menggunakan bentuk dasar berupa garis dan persegi panjang.

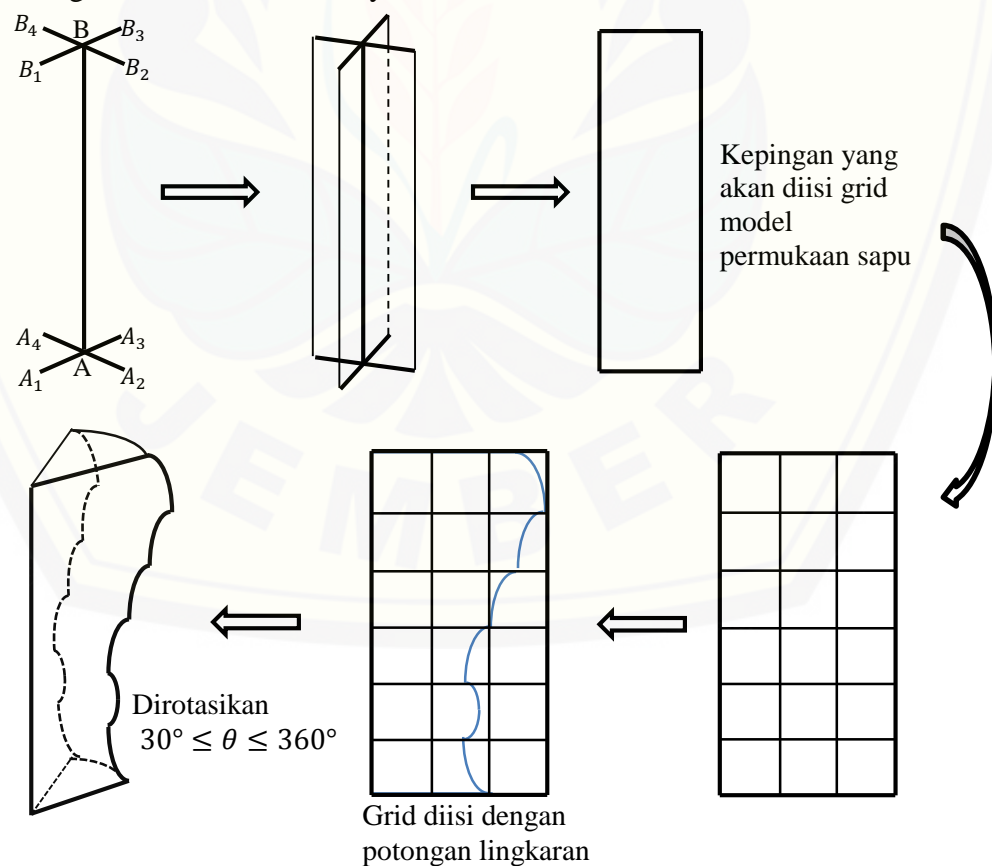
Kelebihan dari model sapu yang akan dikembangkan ini adalah pada bagian tongkat sapu terdapat pegangan untuk menempatkan jari-jari tangan pengguna sehingga lebih mudah dalam menyapu dan tidak licin sebagaimana kebanyakan model sapu yang sudah ada. Terdapat berbagai model relief di bagian tongkat

sehingga menambah keindahan. Kelebihan lain yaitu pada bagian lakop yang memiliki variasi bentuk disesuaikan dengan fungsi sapu dan pada bagian ijuk terdapat model seperti menyilang dan heliks.

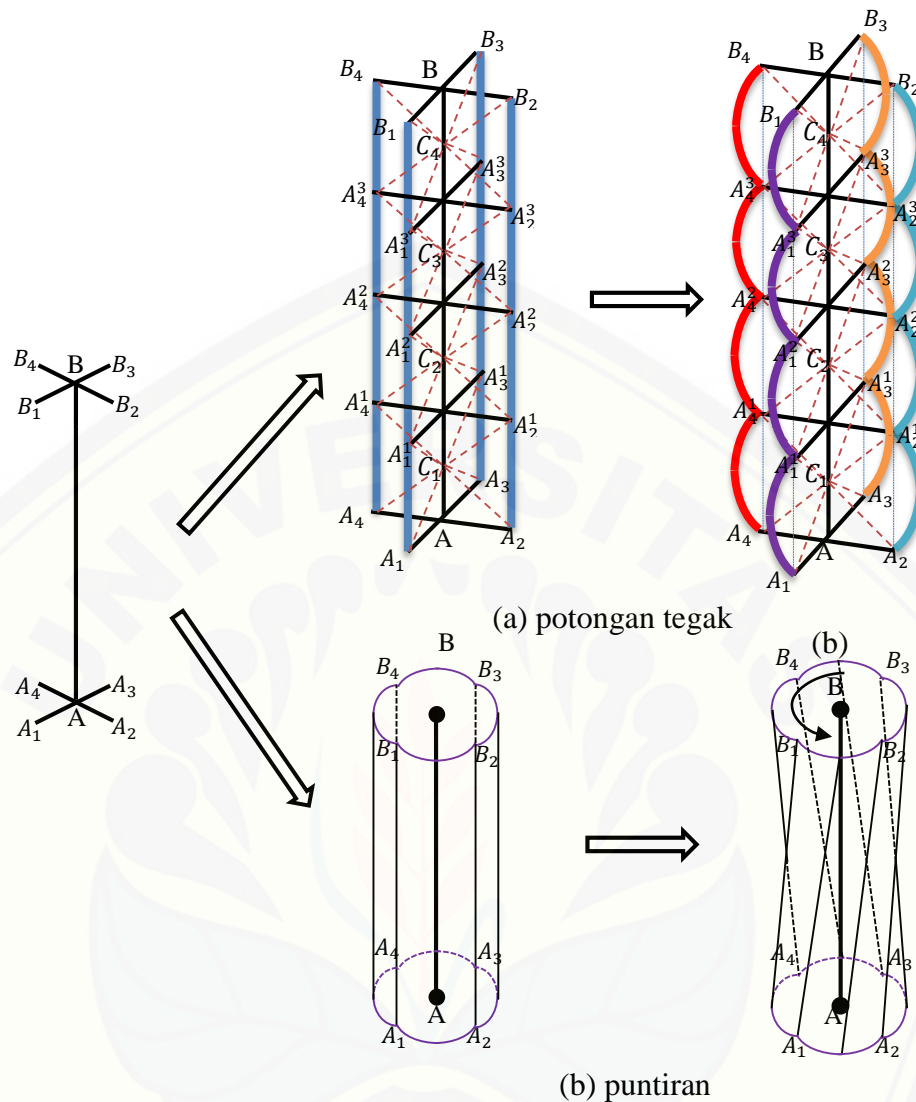
1.2 Rumusan Masalah

Permasalahan yang akan dibahas dalam pemodelan sapu ini meliputi konstruksi bagian tongkat, lakop, dan ijuk. Untuk detailnya dijelaskan sebagai berikut:

1. ditetapkan data penelitian bagian tongkat sapu adalah segmen vertikal \overline{AB} dengan ketinggian $20\text{cm} \leq t \leq 90\text{cm}$ dan tegak lurus dengan segmen horisontal $\overline{A_1A_3}$, $\overline{A_2A_4}$, $\overline{B_1B_3}$, dan $\overline{B_2B_4}$ yang memiliki panjang $2\text{cm} \leq t \leq 20\text{cm}$. Permasalahannya adalah bagaimana mengkonstruksi bagian tongkat sapu, sehingga bentuknya lengkung, kepingan, puntiran, potongan tegak, dan berbagai bentuk variasi lainnya.

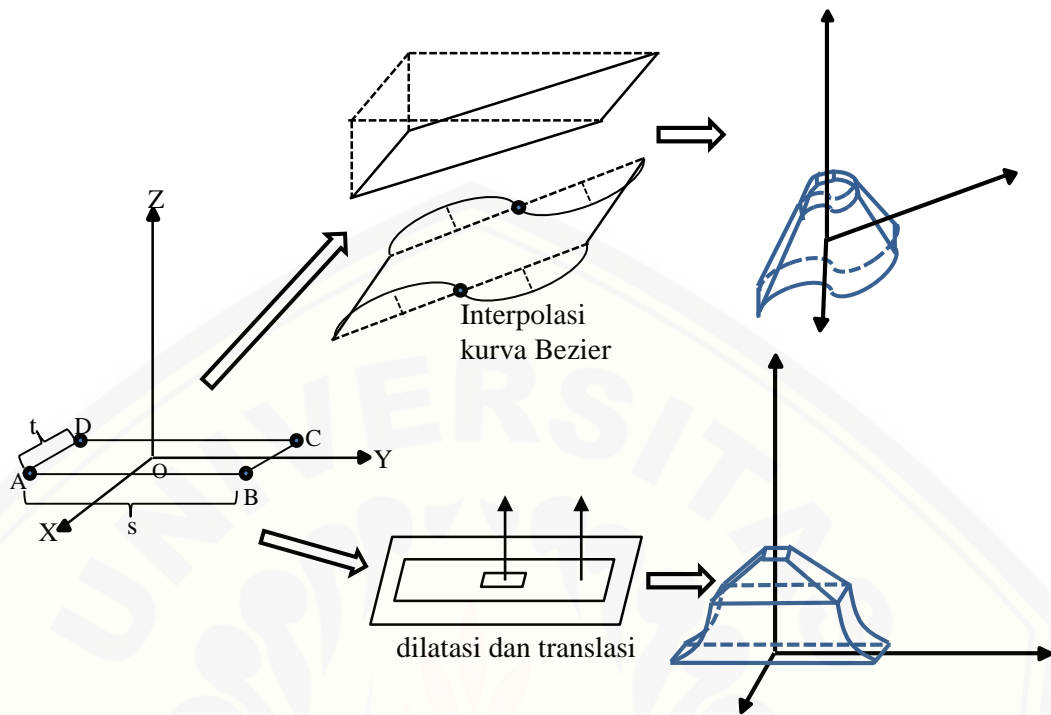


Gambar 1.2.Langkah-langkah modelisasi tongkat sapu

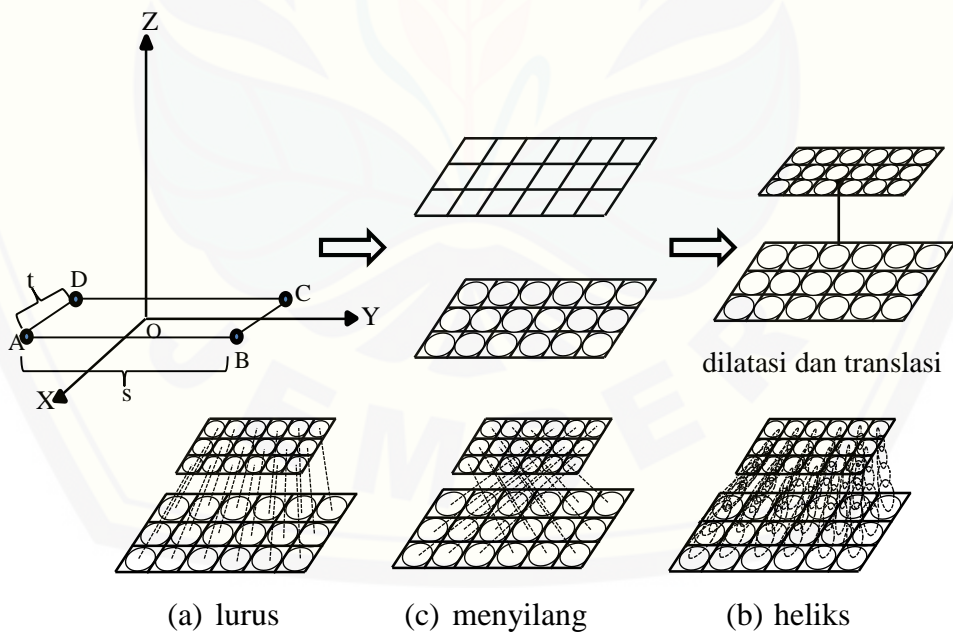


Gambar 1.3 Modelisasi tongkat sapu bentuk potongan tegak dan puntiran

- diketahui sistem koordinat Cartesius XYZ , pada bidang XOY terdapat persegi panjang $ABCD$ bertitik berat di $O(0,0,0)$ dengan titik tengah \overline{AB} berada di sumbu X positif dan titik tengah \overline{CD} berada di sumbu Y negatif. Panjang \overline{AB} dan \overline{CD} masing-masing adalah $2\text{cm} \leq s \leq 30\text{cm}$ dan $2\text{cm} \leq t \leq 6\text{cm}$. Permasalahannya adalah bagaimana mendesain beragam bentuk bagian lakop dan ijuk sapu bersumbu simetris putar sumbu z beralaskan persegi panjang $ABCD$ (Gambar 1.4 dan 1.5).



Gambar 1.4 Modelisasi lakop sapu bentuk persegi panjang



Gambar 1.5 Modelisasi ijuk sapu

1.3 Tujuan Penelitian

Tujuan dari penelitian ini adalah:

1. Mendapatkan prosedur untuk mendesain model tongkat sapu ijuk dengan variasi relief permukaan melalui transformasi benda geometri bidang agar bentuk tongkat tidak monoton dan memberi kemudahan kepada penggunanya.
2. Mendapatkan prosedur untuk mendesain sapu bagian lakop dan ijuk menggunakan benda dasar persegi panjang dan transformasi geometri agar mendapatkan variasi bentuk permukaan lakop dan model ijuk yang beragam.

1.4 Manfaat Penelitian

Adapun manfaat yang dapat diperoleh dalam penelitian ini antara lain:

1. Menambah wawasan dan pengetahuan tentang konstruksi tongkat, lakop, dan ijuk sapu melalui bantuan komputer.
2. Dapat digunakan sebagai alternatif pembuatan tongkat, lakop, dan ijuk sapu yang variatif bentuknya.

BAB 2. TINJAUAN PUSTAKA

Berkaitan dengan beberapa persoalan yang dimaksud untuk mencari solusi permasalahan desain tongkat, lakop, dan ijuk sapu, pada bab ini akan disajikan beberapa teori dasar yang berkaitan dengan prosedur desain tongkat, lakop, dan ijuk sapu. Adapun teori dasar tersebut meliputi kajian tentang penyajian segmen garis, penyajian benda-benda geometri di bidang seperti lingkaran, elips, dan persegi panjang, transformasi bidang di R^3 , kurva hermit, kurva bezier, kurva heliks, serta interpolasi diantara segmen garis dan kurva di ruang. Hal ini bertujuan untuk mempermudah modelisasi dan perangkaian komponen tongkat, lakop, dan ijuk sapu.

2.1 Penyajian Segmen Garis

Ruas (segmen) garis AB dinotasikan \overline{AB} adalah himpunan titik-titik dari garis yang memuat titik A , titik B , dan semua titik diantara titik A dan titik B (Kusno, 2003).

2.1.1 Segmen Garis di Bidang

Misalkan diberikan dua buah titik berbeda di bidang, dengan koordinat masing-masing $A(x_1, y_1)$ dan $B(x_2, y_2)$, maka segmen garis \overline{AB} dapat dinyatakan sebagai tempat kedudukan titik-titik $C(x, y)$ berikut (Gambar 2.1)

$$\overrightarrow{OC} = t\overrightarrow{OB} + (1-t)\overrightarrow{OA} \quad (2.1)$$

dengan $t \in [0,1]$.

Bentuk persamaan (2.1) dapat ditulis menjadi

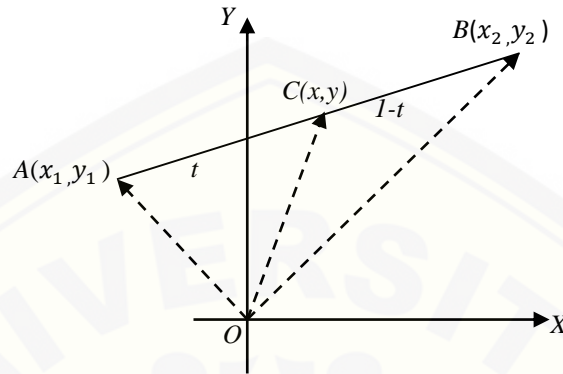
$$\begin{bmatrix} x \\ y \end{bmatrix} = (1-t) \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} + t \begin{bmatrix} x_2 \\ y_2 \end{bmatrix}. \quad (2.2)$$

Sehingga bentuk persamaan parametriknya adalah:

$$\begin{aligned} x(t) &= (1-t)x_1 + tx_2, \\ y(t) &= (1-t)y_1 + ty_2. \end{aligned} \quad (2.3)$$

Jarak d dari A terhadap B merupakan panjang \overline{AB} diformulasikan sebagai

$$d = \overline{AB} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}. \quad (2.4)$$



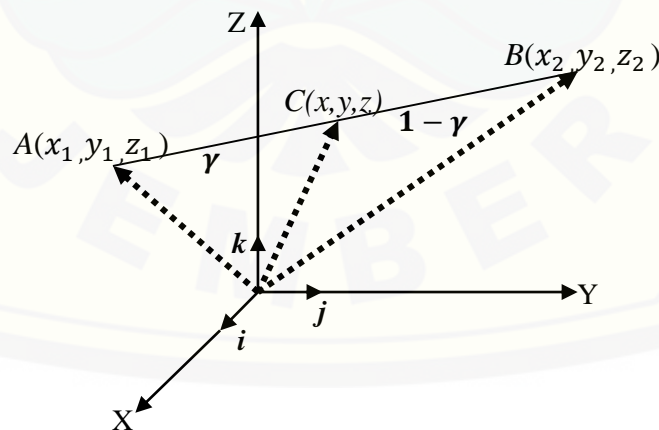
Gambar 2.1 Penyajian segmen garis di bidang

2.1.2 Segmen Garis di Ruang

Misalkan diberikan dua buah titik berbeda di ruang dengan koordinat masing-masing $A(x_1, y_1, z_1)$ dan $B(x_2, y_2, z_2)$, maka segmen garis \overline{AB} dapat didefinisikan secara vektorial sebagai berikut (Gambar 2.2).

$$\overrightarrow{OC} = \gamma \overrightarrow{OB} + (1 - \gamma) \overrightarrow{OA}, \quad (2.5)$$

dengan $\gamma \in [0, 1]$ sebagai variabel parameter dan $C \in \overline{AB}$.



Gambar 2.2 Penyajian segmen garis di ruang

Bentuk persamaan parametrik (2.5) dapat ditulis menjadi

$$\langle x, y, z \rangle = \gamma \langle x_2, y_2, z_2 \rangle + (1 - \gamma) \langle x_1, y_1, z_1 \rangle, \quad (2.6)$$

atau

$$\begin{aligned}x &= (1 - \gamma)x_1 + \gamma x_2, \\y &= (1 - \gamma)y_1 + \gamma y_2, \\z &= (1 - \gamma)z_1 + \gamma z_2.\end{aligned}\tag{2.7}$$

2.2 Penyajian Benda-benda Geometri di Bidang

2.2.1 Lingkaran

Lingkaran merupakan himpunan titik-titik di bidang yang berjarak sama dari suatu titik tetap. Titik tetap ini disebut pusat lingkaran dan jarak yang sama disebut jari-jari.

Misalkan diketahui sebarang titik $A(x,y)$ pada lingkaran yang berpusat di $O(0,0)$ (Gambar 2.3a), maka bentuk persamaan lingkarannya adalah

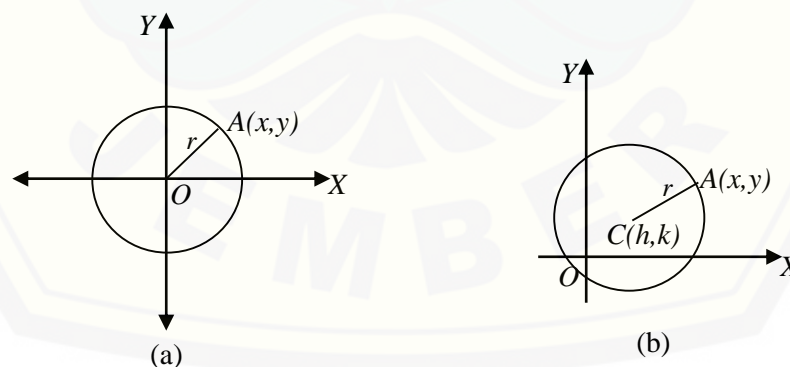
$$\sqrt{x^2 + y^2} = r \text{ atau } x^2 + y^2 = r^2\tag{2.8}$$

dengan r adalah jari-jari lingkaran. Sedangkan lingkaran yang berpusat di $C(h,k)$ dan berjari-jari r (Gambar 2.3b) memiliki persamaan

$$\sqrt{(x - h)^2 + (y - k)^2} = r \text{ atau } (x - h)^2 + (y - k)^2 = r^2.\tag{2.9}$$

Jika Persamaan (2.9) diuraikan diperoleh bentuk

$$x^2 + y^2 - 2hx - 2ky + (h^2 + k^2 - r^2) = 0.\tag{2.10}$$



Gambar 2.3 Penyajian lingkaran

Jadi persamaan umum lingkaran dapat ditulis:

$$x^2 + y^2 + Dx + Ey + F = 0\tag{2.11}$$

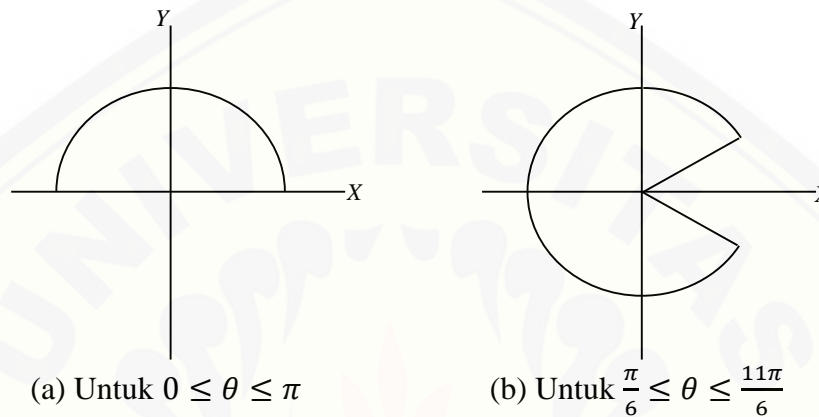
dengan $D = -2h$, $E = -2k$, dan $F = h^2 + k^2 - r^2$.

Selain itu persamaan lingkaran berpusat di $C(h,k)$ dapat dinyatakan dalam bentuk penyajian parametrik:

$$L(\theta) = \langle x_1 + R\cos\theta, y_1 + R\sin\theta \rangle, \quad (2.12)$$

dengan $0 \leq \theta \leq 2\pi$, dan R merupakan jari-jari lingkaran.

Apabila parameter θ pada persamaan (2.12) diberikan nilai dalam interval $\theta_1 \leq \theta \leq \theta_2$, maka diperoleh keratan lingkaran (Gambar 2.4).



Gambar 2.4 Penyajian keratan lingkaran

2.2.2 Elips

Menurut Kusno (2009), elips merupakan himpunan titik-titik yang jumlah jaraknya terhadap dua titik tertentu (fokus elips) besarnya tetap.

Misalkan diberikan dua buah titik sebarang $P(x_p, y_p)$ dan $Q(x_q, y_q)$ yang terletak segaris dan sejajar sumbu OX di bidang XOY . Akan dibangun sebuah elips melalui dua buah titik P dan Q yang berpusat di titik T . Titik T berada di tengah-tengah \overline{PQ} dan sumbu-sumbu elipsnya adalah \overline{PQ} dan \overline{RS} , dimana \overline{RS} tegak lurus \overline{PQ} melalui T .

Melalui Persamaan (2.1) dibangun segmen garis PQ dengan koordinat titik T yang terletak di tengah-tengah segmen garis PQ melalui Persamaan (2.4) diperoleh

$$\overrightarrow{OT} = \frac{1}{2}\overrightarrow{OP} + \frac{1}{2}\overrightarrow{OQ} \quad (2.13)$$

selanjutnya menentukan koordinat titik R yaitu

$$\overrightarrow{OR} = \overrightarrow{OT} + k\mathbf{n}_{\overline{PQ}} \quad (2.14)$$

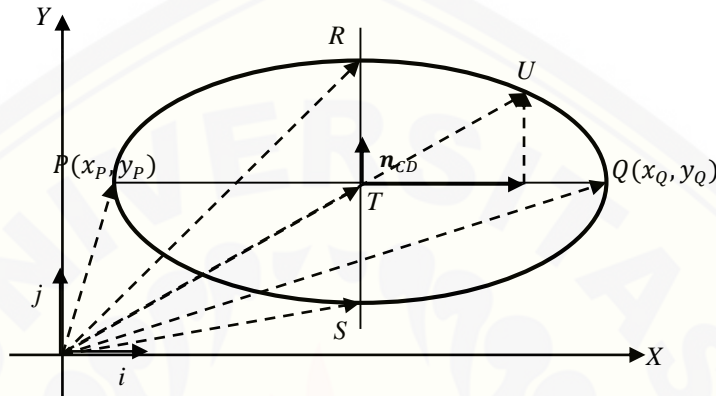
maka dapat dibangun elips melalui persamaan

$$\vec{OU} = \vec{OT} + \gamma_1 \vec{TQ} + \gamma_2 \vec{TR} \tag{2.15}$$

sehingga didapatkan persamaan parametrik sebagai berikut

$$T(t) = (a \cos t, b \sin t) \tag{2.16}$$

dengan $a = |\vec{TQ}|, b = |\vec{TR}|, \gamma_1 = \cos t, \gamma_2 = \sin t$ dan $0 \leq t \leq 2\pi$.



Gambar 2.5 Penyajian elips

2.2.3 Persegi Panjang

Menurut Rich (2004), persegi panjang merupakan jajargenjang yang mempunyai satu sudut siku-siku dimana sudut-sudut yang berurutan adalah suplementer. Sehingga jika salah satu sudutnya merupakan siku-siku, maka sudut yang lainnya juga siku-siku.

Untuk membangun bentuk persegi panjang tahapannya sebagai berikut:

- Ambil dua buah titik sebarang $A(x_A, y_A)$ dan $B(x_B, y_B)$ di bidang XOY .
- Buat segmen garis \overline{AB} dengan Persamaan (2.1).
- Hitung koordinat titik C melalui kondisi : $\vec{OC} = \vec{OB} + k\mathbf{n}_{AB}$ dengan \mathbf{n}_{AB} merupakan normal \overline{AB} dalam bentuk

$$\mathbf{n}_{AB} = \frac{\langle (y_B - y_A) - (x_B - x_A) \rangle}{\sqrt{(x_B - x_A)^2 + (y_B - y_A)^2}} \text{ dan } k \text{ ditetapkan, sehingga :}$$

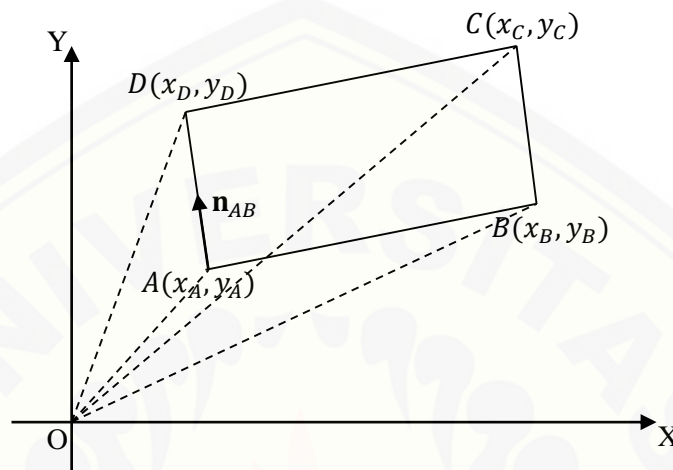
$$\vec{OC} = \vec{OB} + k\mathbf{n}_{AB} = \begin{pmatrix} x_B \\ y_B \end{pmatrix} + k \left(\frac{\langle (y_B - y_A) - (x_B - x_A) \rangle}{\sqrt{(x_B - x_A)^2 + (y_B - y_A)^2}} \right). \tag{2.17}$$

- Hitung koordinat titik D dengan persamaan:

$\vec{OD} = \vec{OA} + \vec{BC}$, sehingga

$$(x_D, y_D) = (x_A + x_C - x_B, y_A + y_C - y_B). \quad (2.18)$$

- e. Bangun segmen garis \overline{AD} dan \overline{CD} melalui Persamaan (2.1) sehingga terbentuk persegi panjang $ABCD$.



Gambar 2.6 Penyajian persegi panjang

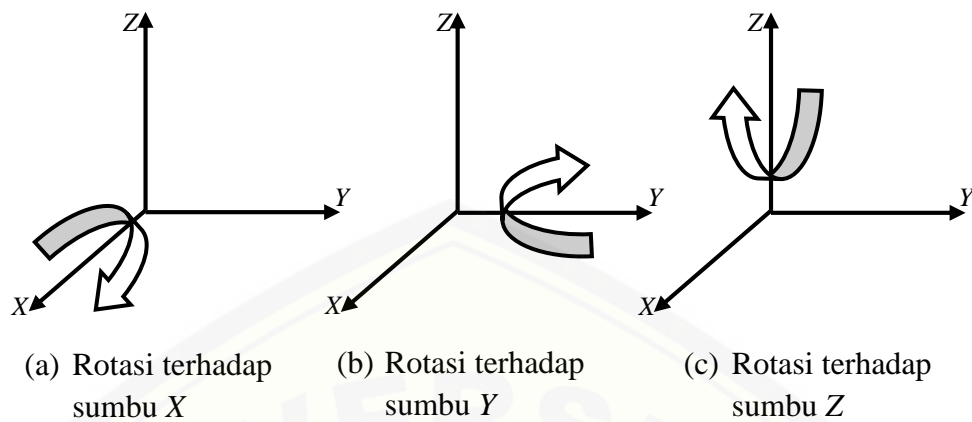
2.3 Transformasi Bidang di R^3

Transformasi terdiri atas rotasi (perputaran), dilatasi (penskalaan), translasi (pergeseran), dan refleksi (pencerminan). Sehubungan dengan penelitian ini hanya dibahas masalah rotasi, dilatasi, dan translasi.

2.3.1 Rotasi (Perputaran)

Rotasi dalam R^3 dikenal dua sistem koordinat, yaitu sistem koordinat tangan kiri dan sistem koordinat tangan kanan. Pada sistem koordinat tangan kiri, rotasi bersudut positif dinyatakan searah putaran jarum jam. Sedangkan pada sistem koordinat tangan kanan, rotasi bersudut positif dinyatakan berlawanan arah putaran jarum jam. Rotasi bidang geometri ditentukan oleh titik pusat, besar sudut, dan arah rotasi (Sukirman, 1993).

Rotasi pada sistem koordinat tangan kiri di sumbu XYZ digambarkan sebagai berikut (Gambar 2.7).



Gambar 2.7 Ilustrasi rotasi pada sistem koordinat tangan kiri

Persamaan rotasi secara umum adalah $QP=PR$, dengan P adalah posisi titik sebelum dirotasi, Q adalah posisi titik setelah dirotasi, dan R adalah matriks rotasi. Jika diketahui θ adalah besarnya sudut rotasi bertitik pangkal rotasi di $O(0,0,0)$, maka rotasi terhadap masing-masing sumbu dapat dituliskan dalam bentuk matriks yaitu matriks koefisien yang bersesuaian dengan transformasi rotasi.

Rotasi terhadap sumbu X:

$$[X_q \ Y_q \ Z_q] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & \sin \theta \\ 0 & -\sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} X_p \\ Y_p \\ Z_p \end{bmatrix} \quad (2.19)$$

Rotasi terhadap sumbu Y:

$$[X_q \ Y_q \ Z_q] = \begin{bmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{bmatrix} \begin{bmatrix} X_p \\ Y_p \\ Z_p \end{bmatrix} \quad (2.20)$$

Rotasi terhadap sumbu Z:

$$[X_q \ Y_q \ Z_q] = \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_p \\ Y_p \\ Z_p \end{bmatrix}. \quad (2.21)$$

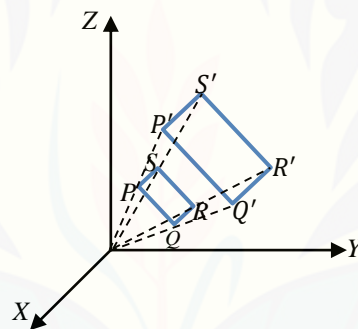
2.3.2 Dilatasi (Penskalaan)

Menurut Kusno (2009), transformasi dilatasi yang memetakan titik $P(x, y, z)$ ke $P'(x', y', z')$ diformulasikan sebagai:

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} k_1 & 0 & 0 \\ 0 & k_2 & 0 \\ 0 & 0 & k_3 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} k_1 x \\ k_2 y \\ k_3 z \end{bmatrix}, \quad (2.22)$$

dengan $k_1, k_2, k_3 \in \text{real}$, dimana harga k_1 menyajikan skala ke arah sumbu X , k_2 ke arah sumbu Y , dan k_3 menyajikan skala ke arah sumbu Z . Jika $k_1 = k_2 = k_3$, maka peta obyek yang didapat sebangun dengan obyek aslinya (mungkin diperbesar, diperkecil atau tetap).

Misalkan persegi panjang $PQRS$ dengan titik-titik sudut $P(x_1, y_1, z_1), Q(x_2, y_2, z_2), R(x_3, y_3, z_3)$ dan $S(x_4, y_4, z_4)$ dilatasi dengan faktor pengali $k > 1$, sehingga didapatkan persegi panjang bayangan $P'Q'R'S'$ dengan titik-titik sudut $P'(kx_1, ky_1, kz_1), Q'(kx_2, ky_2, kz_2), R'(kx_3, ky_3, kz_3)$, dan $S'(kx_4, ky_4, kz_4)$ (Gambar 2.8).



Gambar 2.8 Dilatasi dengan $k > 1$

2.3.3 Translasi (Pergeseran)

Translasi adalah perpindahan kedudukan sebarang titik dengan penambahan besaran pada arah sumbu X , Y , dan Z (Setyo Budhi, 1995). Translasi bersifat mempertahankan bentuk dan ukuran obyek.

Secara umum translasi dapat dinyatakan oleh persamaan $Q = P + K$, dimana P adalah posisi titik awal sebelum translasi, Q adalah posisi titik setelah translasi, dan K adalah besarnya pergeseran ke arah sumbu X , Y , dan Z . Persamaan translasi dalam bentuk matriks dapat ditulis sebagai berikut:

$$\begin{bmatrix} X_q & Y_q & Z_q \end{bmatrix} = \begin{bmatrix} X_p \\ Y_p \\ Z_p \end{bmatrix} + \begin{bmatrix} X_k \\ Y_k \\ Z_k \end{bmatrix}. \quad (2.23)$$

Sedangkan persamaan dalam bentuk koordinat adalah

$$(X_q, Y_q, Z_q) = (X_p + X_k, Y_p + Y_k, Z_p + Z_k). \quad (2.24)$$

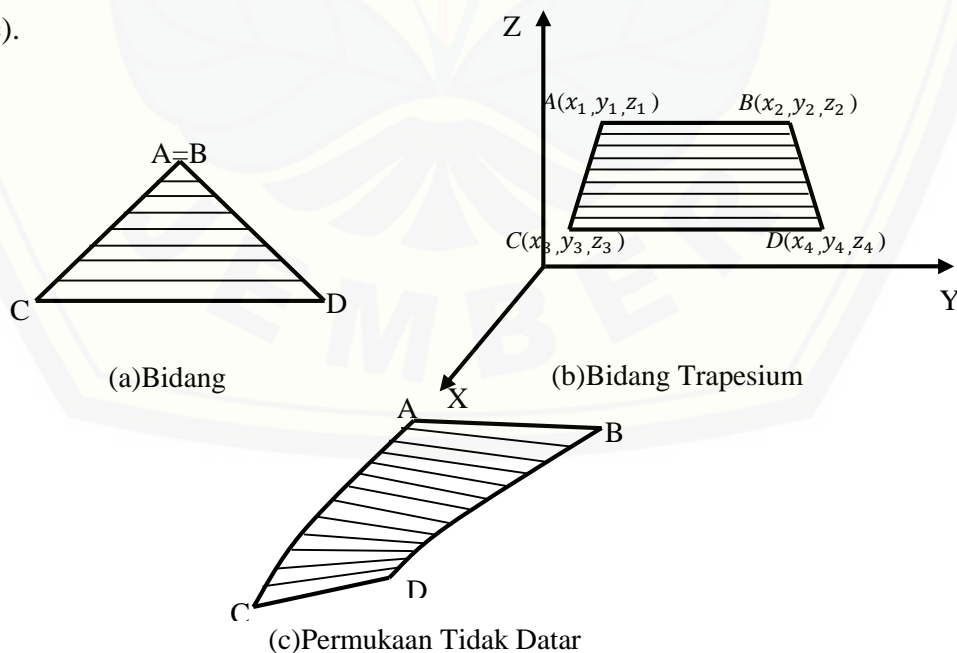
2.4 Interpolasi diantara Segmen Garis dan Kurva di Ruang

Misalkan terdapat dua segmen garis \overline{AB} dan \overline{CD} didefinisikan masing-masing oleh $A(x_1, y_1, z_1)$, $B(x_2, y_2, z_2)$, $C(x_3, y_3, z_3)$ dan $D(x_4, y_4, z_4)$ dalam bentuk parametrik $I_1(u)$ dan $I_2(u)$, maka dari persamaan $\langle x, y, z \rangle = t\langle x_2, y_2, z_2 \rangle + (1 - t)\langle x_1, y_1, z_1 \rangle$ untuk membangun permukaan parametrik hasil interpolasi linier kedua segmen garis tersebut diformulasikan sebagai berikut:

$$S(u, v) = (1 - v)I_1(u) + vI_2(u), \quad (2.25)$$

dengan $0 \leq u \leq 1$ dan $0 \leq v \leq 1$, u dan v sebagai parameter.

Terdapat beberapa kasus khusus interpolasi linier kedua garis tersebut. Jika $A=B$ maka hasil interpolasi persamaan (2.25) akan menghasilkan bidang segitiga (Gambar 2.9a). Sedangkan jika $\overline{AB} \parallel \overline{CD}$ maka secara umum akan membentuk bidang segiempat (Gambar 2.9b). Jika bidang tersebut dibentuk dari interpolasi dua garis yang bersilangan maka menghasilkan permukaan tidak datar (dapat melengkung ataupun terjadi puntiran di sebagian permukaan tersebut) (Gambar 2.9c).

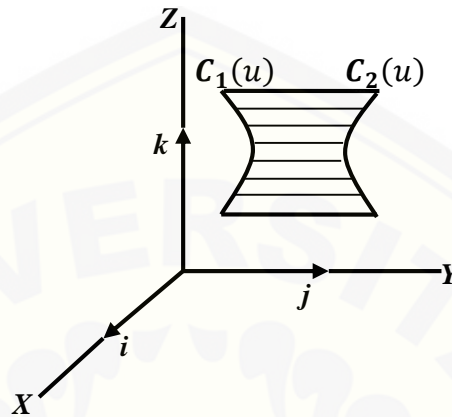


Gambar 2.9 Contoh kasus khusus interpolasi linier dan segmen garis

Di lain pihak dapat dibangun permukaan lengkung hasil interpolasi kurva ruang melalui persamaan berikut:

$$S(u, v) = (1 - v)C_1(u) + vC_2(u), \tag{2.26}$$

dengan $C_1(u)$ dan $C_2(u)$ merupakan kurva batas (Gambar 2.11).



Gambar 2.10 Interpolasi linier pada kurva

2.5 Kurva Hermit Kuadratik

Kurva hermit kuadratik dapat dinyatakan sebagai berikut (Kusno,2009) :

$$P(u) = P(0)H_1(u) + P(1)H_2(u) + P^u(1)H_3(u) \tag{2.27}$$

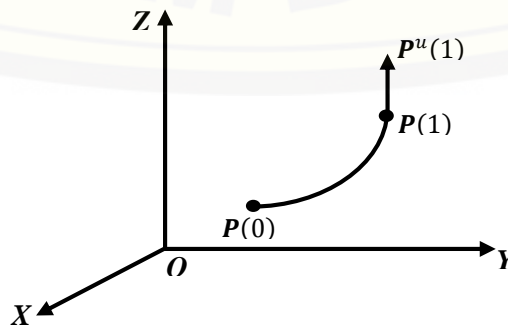
dengan

$$\begin{aligned} H_1(u) &= u^2 - 2u + 1, \\ H_2(u) &= -u^2 + 2u, \\ H_3(u) &= u^2 - u. \end{aligned} \tag{2.28}$$

$P(0)$ = titik awal kurva

$P(1)$ = titik akhir kurva

$P^u(1)$ = vektor singgung di $P(1)$ dengan $0 \leq u \leq 1$.



Gambar 2.11 Kurva Hermit kuadratik

2.6 Kurva Heliks

Heliks dapat didefinisikan melalui persamaan parametrik sebagai berikut:

$$\begin{aligned}x(t) &= r \cos(t), \\y(t) &= r \sin(t), \\z(t) &= t,\end{aligned}\tag{2.29}$$

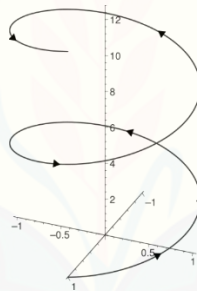
dengan $a \leq t \leq b$.

Kurva heliks merupakan kurva berbentuk spiral yang terletak pada silinder $x^2 + y^2 = r^2$ karena $x^2 + y^2 = r^2 \cos^2 t + r^2 \sin^2 t = r^2$. Semua titik (x, y, z) dari kurva heliks terletak tepat di atas titik $(x, y, 0)$ dari lingkaran tersebut di bidang XOY (Gambar 2.12).

Persamaan vektor heliks dapat dinyatakan sebagai berikut:

$$\mathbf{H}_L(t) = r \cos(t) \mathbf{i} + r \sin(t) \mathbf{j} + r t \mathbf{k} = (r \cos t, r \sin(t), t)\tag{2.30}$$

dengan $0 \leq t \leq 2\pi$.



Gambar 2.12 Kurva Heliks

2.7 Kurva Bezier

Kurva Bezier merupakan kurva berparameter yang sering digunakan dalam grafika komputer dan bidang yang berkaitan. Generalisasi dari kurva Bezier ke dimensi yang lebih tinggi disebut permukaan Bezier. Menurut Kusno (2009), kurva Bezier derajat- n $C(u)$ dinyatakan dalam bentuk parametrik berikut:

$$C(u) = \sum_{i=0}^n \mathbf{P}_i B_i^n(u), 0 \leq u \leq 1\tag{2.31}$$

dengan:

$$B_i^n(u) = C_i^n (1-u)^{n-1} u^i,$$

$$C_i^n = \frac{n!}{i!(n-i)!}$$

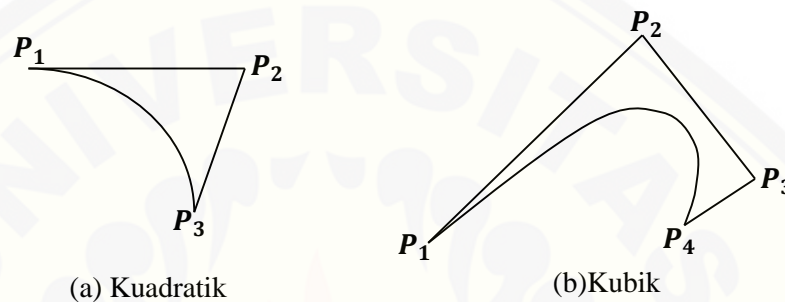
P_i = koefisien geometri / titik kontrol kurva $C(u)$.

Jika $n=2$, akan dihasilkan kurva Bezier kuadratik dengan persamaan parametric (Gambar 2.12a):

$$C(u) = (1-u)^2 P_0 + 2(1-u)(u)P_1 + u^2 P_2,$$

sedangkan untuk $n=3$ didapatkan empat titik kontrol yaitu P_0, P_1, P_2 , dan P_3 sehingga persamaan parametrik kurva Bezier kubiknya adalah (Gambar 2.9b):

$$C(u) = (1-u)^3 P_0 + 3(1-u)^2(u)P_1 + 3(1-u)u^2 P_2 + u^3 P_3$$



Gambar 2.13 Kurva Bezier

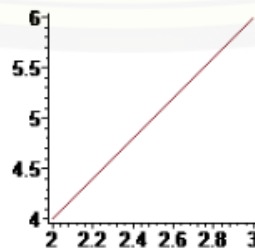
2.6 Konstruksi Objek Dasar pada Program Maple 18

Berkaitan dengan teori diatas akan disajikan beberapa contoh pemrograman untuk mengkonstruksi objek geometri menggunakan *software* Maple 18. Adapun contoh programnya sebagai berikut.

a. Mengkonstruksi segmen garis di bidang

Konstruksi segmen garis AB di bidang dengan titik-titik ujung $A(2,4)$ dan $B(3,6)$ dapat disimulasikan menggunakan Maple 18 melalui *script* program fungsi parametrik sebagai berikut:

```
>AB:=plot([2*(1-t)+3*t, 4*(1-t)+6*t, t=0..1]):
```

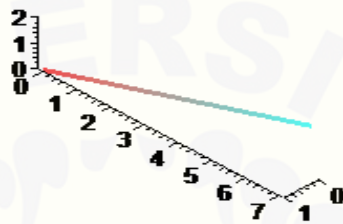


Gambar 2.14 Penyajian segmen garis di bidang

b. Mengkonstruksi segmen garis di ruang

Untuk membuat segmen garis digunakan persamaan (2.6) dengan memberikan nilai (x_1, y_1, z_1) dan (x_2, y_2, z_2) sebagai posisi titik ujung segmen garis di ruang. Misalkan akan dibuat suatu segmen garis a dengan titik-titik ujung $A(1,0,0)$ dan $B(0,7,2)$. Berikut ini merupakan *script* program Maple 18.

```
spacecurve([(1-t)*1+t*0, (1-t)*0+t*7, (1-t)*0+t*2, t=0..1],
,thickness=3);
```

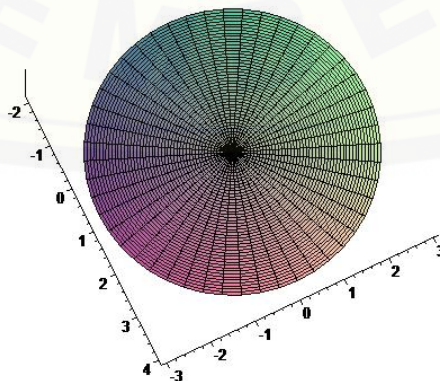


Gambar 2.15 Penyajian segmen garis di ruang

c. Mengkonstruksi bidang lingkaran

Bidang lingkaran dapat disajikan dengan menggunakan program Maple 18. Untuk membuat bidang lingkaran dapat menggunakan persamaan (2.9) dengan memberikan nilai pusat dan jari-jarinya. Misalkan dibangun lingkaran berpusat di titik $A(1,0,0)$, berjari-jari 3 satuan dengan $0 \leq t \leq 2\pi$ melalui perintah *script* program sebagai berikut:

```
>plot3d([r*3*cos(t)+1, r*3*sin(t)+0, 0], r=0..1, t=0..2*Pi,
scaling=constrained, thickness=1);
```

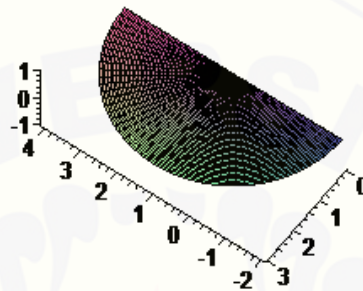


Gambar 2.16 Penyajian lingkaran

d. Mengkonstruksi potongan bidang lingkaran

Fungsi parametrik untuk membangun potongan lingkaran berpusat di titik $O(1,0,0)$, berjari-jari 3 satuan dan $0 \leq t \leq \pi$ dengan Maple 18 dapat melalui perintah *script* program sebagai berikut:

```
>plot3d([r*3*cos(t)+1, r*3*sin(t)+0, 0], r=0..1, t=0..Pi, scaling=constrained, thickness=1);
```

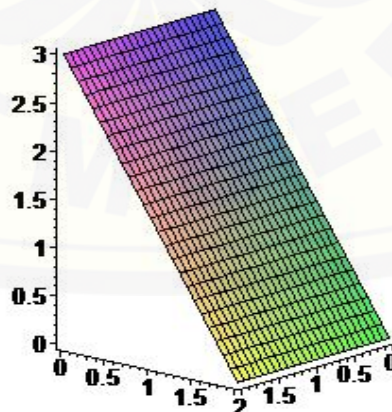


Gambar 2.17 Penyajian potongan lingkaran

e. Mengkonstruksi persegi panjang

Persegi panjang dapat disajikan dengan menggunakan program Maple. Misalkan dibangun persegi panjang (Gambar 2.18) dengan titik sudut-titik sudut $P(2,2,0)$, $Q(0,2,0)$, $R(2,0,3)$, dan $S(0,0,3)$ maka *script* program Maple 18 adalah:

```
Plot3d([(1-v)*(2-2*u)+v*(2-2*u), (1-v)*2+v*0, (1-v)*0+v*3], u=0..1, v=0..1, scaling=constrained);
```

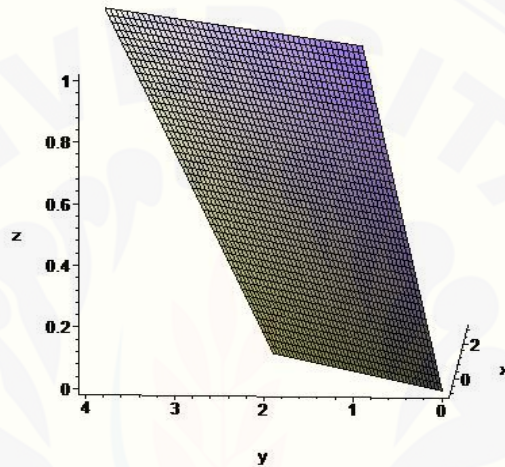


Gambar 2.18 Penyajian persegi panjang

f. Mengkonstruksi Interpolasi

Konstruksi interpolasi garis AB dan garis CD dengan $A(-1,0,0)$, $B(1,2,0)$, $C(1,1,1)$, dan $D(3,4,1)$ dapat disimulasikan dengan menggunakan Maple 18 melalui perintah *script* program sebagai berikut:

```
> x1:=t*(-1)+(1-t)*1: y1:=t*0+(1-t)*2: z1:=t*0+(1-t)*0:
> x2:=t*1+(1-t)*3: y2:=t*1+(1-t)*4: z2:=t*1+(1-t)*1:
> plot3d([v*x1+(1-v)*x2, v*y1+(1-v)*y2, v*z1+(1-v)*z2],
t=0..1, v=0..1, labels=[x, y, z]);
```

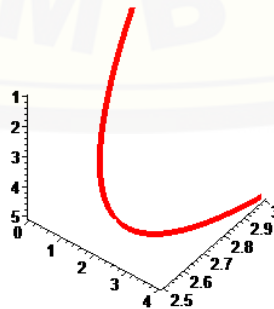


Gambar 2.19 Penyajian interpolasi

g. Mengkonstruksi Kurva Hermit

Contoh *script* program Maple 18 untuk membangun kurva Hermit ditunjukkan pada Gambar 2.20 berikut.

```
spacecurve([(u+3)*h1+(u+2)*h2+(1*h3), (u+0)*h1+(u+3)*h2+
(1*h3), (u+1)*h1+(u+4)*h2+(1*h3)], u=0..1);
```

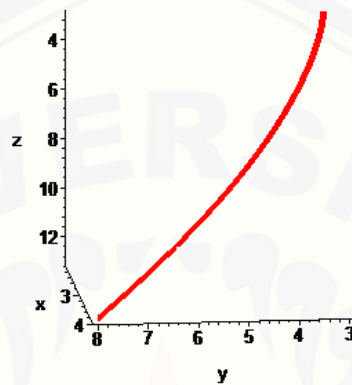


Gambar 2.21 Penyajian kurva Hermit

h. Mengkonstruksi Kurva Bezier

Contoh *script* program Maple 18 untuk membangun kurva Bezier ditunjukkan pada Gambar 2.21 berikut.

```
>> spacecurve([4*(1-t)^2+3*2*(1-t)*t+t^2*2, 8*(1-t)^2+3*2*(1-t)*t+t^2*3, 7*(1-t)^2+3*2*(1-t)*t+t^2*3], t=0..1, color=red, thickness=5, labels=[x, y, z]):
```



Gambar 2.21 Penyajian kurva Bezier

BAB 3. METODE PENELITIAN

Berdasarkan rumusan masalah pada subbab 1.2 dan hasil kajian tinjauan pustaka pada Bab 2, untuk penyelesaian permasalahan tersebut diuraikan langkah-langkah penelitian sebagai berikut:

- a. Memodelisasikan benda dasar geometri untuk membangun bagian-bagian komponen sapu yaitu bagian tongkat, lakop dan ijuk. Masing-masing bagian akan dilakukan modelalisasi dengan urutan sebagai berikut:

1. Bagian Tongkat

Mendesain bagian tongkat sapu melalui pola lengkung vertikal, pola potongan tegak kasus cekung cembung dan kasus kepingan, puntiran dengan cara menetapkan data penelitian berupa segmen vertikal dengan ketinggian t dan 4 segmen horisontal membentuk persegi pada titik ujungnya. Kemudian membentuk permukaan tongkat menggunakan kurva Hermit, potongan lingkaran dan elips berdasarkan titik-titik pada data tersebut. Menginterpolasi pasangan-pasangan kurva secara berurutan pada masing-masing pola.

2. Bagian Lakop

Mendesain lakop sapu melalui interpolasi kurva Bezier untuk mendapatkan pola lengkung horisontal maupun vertikal dengan cara menetapkan persegi panjang tegak lurus sumbu OZ kemudian mentranslasi dan mendilatasi persegi panjang tersebut searah sumbu OZ dengan jarak dan faktor pengali tertentu. Menginterpolasi persegi panjang sebelum dan sesudah translasi secara linier.

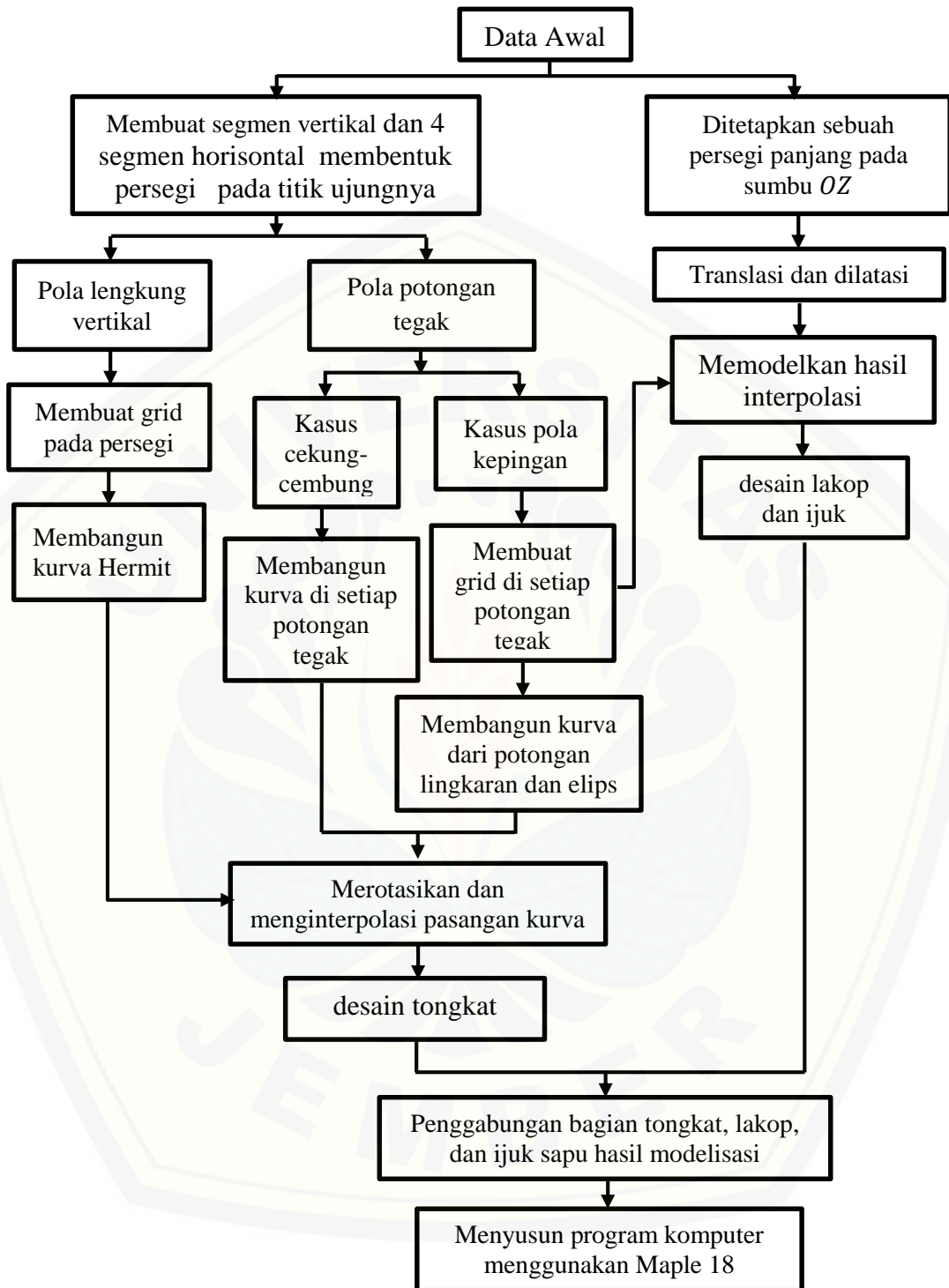
3. Bagian Ijuk

Mendesain ijuk melalui persegi panjang dengan cara translasi dan dilatasi seperti pada lakop kemudian menginterpolasikannya secara lurus, menyilang dan melingkar/heliks.

- b. Penggabungan bagian tongkat, lakop, dan ijuk sapu hasil modelisasi kegiatan (a) dengan menetapkan satu sumbu penggabungan untuk mengontrol kesimetrisan dan ketinggian benda yang perlu digabungkan.
- c. Menyusun program komputer menggunakan Maple 18.

Untuk lebih jelasnya mengenai metode penelitian tersebut dapat dilihat pada skema di halaman berikutnya.





Gambar 3.1 Skema metode penelitian

BAB 5. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian dan pembahasan, diperoleh kesimpulan sebagai berikut.

1. Untuk mendesain beragam bentuk sapu yang terdiri dari 3 komponen yaitu tongkat, lakop, dan ijuk, prosedurnya sebagai berikut. Pertama, membangun bentuk permukaan tongkat sapu dari segmen vertikal dan persegi di kedua ujungnya melalui grid persegi dan potongan tegak dari segmen vertikal, kemudian menginterpolasi pasangan kurva secara berurutan. Kedua, mengkonstruksi bagian lakop melalui translasi dan dilatasi persegi panjang yang dilengkungkan dengan kurva Bezier, kemudian menginterpolasikan persegi panjang sebelum dan sesudah translasi. Ketiga, mengkonstruksi ijuk sapu dari persegi panjang dasar lakop, kemudian menginterpolasikan titik-titik didalam grid persegi panjang.
2. Untuk merangkai sapu ijuk secara utuh diperlukan unsur kesimetrisan melalui satu sumbu pemodelan, prosedurnya sebagai berikut. Pertama, membagi sumbu menjadi beberapa segmen. Kedua, mengisi setiap bagian segmen tersebut dengan bentuk dasar komponen sapu sehingga menghasilkan model sapu yang bervariasi.

5.2 Saran

Pada skripsi ini telah diperkenalkan prosedur modelisasi sapu menggunakan segmen garis vertikal yang tegak lurus dengan persegi dan persegi panjang untuk menghasilkan bentuk yang simetris dan bervariasi. Diharapkan untuk penelitian ke depan metode ini dapat dikembangkan lagi dengan menggunakan benda geometri bidang lainnya seperti segitiga, lingkaran, dan

trapesium atau benda geometri ruang seperti keratan bola, elipsoida, hiperboloida, dan lain sebagainya.



DAFTAR PUSTAKA

- Abrina, D.Q. 2015. *Modelisasi Bros dengan Lingkaran dan Persegi Panjang*. Artikel. <http://repository.unej.ac.id/123456789/65863>. [24 Mei 2016].
- Arinda, D. 2012. *Konstruksi Vas Bunga melalui Penggabungan Beberapa Benda Geometri Ruang*. Artikel. <http://repository.unej.ac.id/handle/123456789/13418>. [21 Mei 2016].
- Budhi, W.S. 1995. *Aljabar Linier*. Jakarta: Gramedia.
- Kusno. 2002. *Geometri Rancang Bangun Studi Aljabar Vektor Garis, Lingkaran, dan Elips*. Jember: Jurusan Matematika Fakultas MIPA Universitas Jember.
- Kusno. 2003. *Diktat Kuliah Geometri*. Jember: Jurusan Matematika Fakultas MIPA Universitas Jember.
- Kusno. 2009. *Geometri Rancang Bangun Studi Tentang Desain dan Pemodelan Benda dengan Kurva dan Permukaan Berbantu Komputer*. Jember: Jurusan Matematika Fakultas MIPA Universitas Jember.
- Rich, B. 2004. *Geometry Schaum's easy Outlines*. Jakarta: Erlangga.
- Sukirman. 1993. *Geometri Analitik Bidang dan Ruang*. Jakarta: Universitas Terbuka, Depdikbud.

LAMPIRAN

A. Lampiran Tongkat Sapu

A.1. Pola Lengkung Vertikal

```

> restart:
with (plots):
with (plottools):
> panjang:=3; tinggi:=10:
n:=panjang/6: t0:=0; t1:=tinggi:
k:=4:

> #grid persegi
grs1:=plot3d([(v)*((u)*(-panjang/2)+(1-u)*(-panjang/2))+(1-v)*((u)
*(panjang/2)+(1-u)*(panjang/2)), (v)*((u)*(panjang/2)+(1-u)*(panja
ng/2))+(1-v)*((u)*(panjang/2)+(1-u)*(panjang/2)), t0], u=0..1, v=0..
1, color=blue):
grs2:=translate(grs1, 0, -1*n, 0):
grs3:=translate(grs1, 0, -2*n, 0):
grs4:=translate(grs1, 0, -3*n, 0):
grs5:=translate(grs1, 0, -4*n, 0):
grs6:=translate(grs1, 0, -5*n, 0):
grs7:=translate(grs1, 0, -6*n, 0):
grs11:=plot3d([(v)*((u)*(panjang/2)+(1-u)*(panjang/2))+(1-v)*((u)
*(panjang/2)+(1-u)*(panjang/2)), (v)*((u)*(-panjang/2)+(1-u)*(-
panjang/2))+(1-v)*((u)*(panjang/2)+(1-u)*(panjang/2)), t0], u=0..1,
v=0..1, color=blue):
grs12:=translate(grs11, -1*n, 0, 0):
grs13:=translate(grs11, -2*n, 0, 0):
grs14:=translate(grs11, -3*n, 0, 0):
grs15:=translate(grs11, -4*n, 0, 0):
grs16:=translate(grs11, -5*n, 0, 0):
grs17:=translate(grs11, -6*n, 0, 0):
grs111:=spacecurve([(1-v)*0+v*0, (1-v)*0+v*0, (1-v)*(5*t1)+v*(0*t1)]
, v=0..1, thickness=1, color=black):
grs112:=spacecurve([(1-v)*k+v*(-k), (1-v)*0+v*0, (1-v)*0+v*0], v=0..1
, thickness=1, color=black):
grs113:=spacecurve([(1-v)*0+v*0, (1-v)*k+v*(-k), (1-v)*0+v*0], v=0..1
, thickness=1, color=black):
a:=display([grs1, grs2, grs3, grs4, grs5, grs6, grs7, grs11, grs12, grs13, grs
14, grs15, grs16, grs17]):
b:=translate(a, 0, 0, t1):
display([a], style=line, axes=frame, scaling=constrained, lightmodel=lig
ht3):
display([a, b], style=line, axes=frame, scaling=constrained, lightmodel=1
ight3):

> #kurva pola bunga pada grid persegi
a1:=spacecurve([n*sqrt(2)*cos(u*Pi/2+Pi*3/4)-n, n*sqrt(2)*sin(u*Pi
/2+Pi*3/4)+n, t0], [n*sqrt(2)*cos(u*Pi/2+Pi*3/4)-n, (n*sqrt(2)*sin(u*
Pi/2+Pi*3/4)-n), t0], u=0..1, axes=frame, scaling=constrained, light
model=light3, thickness=3, color=red):
a2:=rotate(a1, 0, 0, Pi/2):
a3:=rotate(a1, 0, 0, -Pi/2):

```

```

a4:=rotate(a1,0,0,Pi):
a5:=display([a1,a2,a3,a4]):
a6:=translate(a5,0,0,t1):
display([a,a1]):
display([a,a5]):
display([a,a5,b,a6]):

>#interpolasi lengkung vertikal pola bunga
a11:=plot3d({[n*sqrt(2)*cos(u*Pi/2+Pi*3/4)-n,n*sqrt(2)*sin(u*Pi/2+Pi*3/4)+n,v*t0+(1-v)*t1],[n*sqrt(2)*cos(u*Pi/2+Pi*3/4)-n,(n*sqrt(2)*sin(u*Pi/2+Pi*3/4)-n),v*t0+(1-v)*t1]},v=0..1,u=0..1,axes=frame,scaling=constrained,lightmodel=light3,thickness=3):
a12:=rotate(a11,0,0,Pi/2):
a13:=rotate(a11,0,0,-Pi/2):
a14:=rotate(a11,0,0,Pi):
display([a11,a12,a13,a14,grs111,grs112,grs113],labels=[x,y,z]):

>#kurva pola sayap pada grid persegi
b1:=spacecurve({[n*cos(u*Pi/2+Pi*2/4)-2*n,n*sin(u*Pi/2+Pi*2/4)+n,t0],[2*n*cos(u*Pi/2+Pi)-n,(n*sin(u*Pi/2+Pi)+n),t0],[n*cos(u*Pi/2+Pi*2/4)-2*n,-(n*sin(u*Pi/2+Pi*2/4)+n),t0],[2*n*cos(u*Pi/2+Pi)-n,-(n*sin(u*Pi/2+Pi)+n),t0]},u=0..1,style=line,axes=frame,scaling=constrained,lightmodel=light3,thickness=3,color=red):
b2:=rotate(b1,0,0,Pi/2):
b3:=rotate(b1,0,0,-Pi/2):
b4:=rotate(b1,0,0,Pi):
b5:=display([b1,b2,b3,b4]):
b6:=translate(b5,0,0,t1):
display([a,b1]):
display([a,b5]):
display([a,b5,b,b6]):

>#interpolasi lengkung vertikal pola sayap
b11:=plot3d({[n*cos(u*Pi/2+Pi*2/4)-2*n,n*sin(u*Pi/2+Pi*2/4)+n,v*t0+(1-v)*t1],[2*n*cos(u*Pi/2+Pi)-n,(n*sin(u*Pi/2+Pi)+n),v*t0+(1-v)*t1],[n*cos(u*Pi/2+Pi*2/4)-2*n,-(n*sin(u*Pi/2+Pi*2/4)+n),v*t0+(1-v)*t1],[2*n*cos(u*Pi/2+Pi)-n,-(n*sin(u*Pi/2+Pi)+n),v*t0+(1-v)*t1]},v=0..1,u=0..1,style=patchngrid,axes=frame,scaling=constrained,lightmodel=light3,thickness=3):
b12:=rotate(b11,0,0,Pi/2):
b13:=rotate(b11,0,0,-Pi/2):
b14:=rotate(b11,0,0,Pi):
display([b11,b12,b13,b14]):

>#kurva pola jaring pada grid persegi
c1:=spacecurve({[n*cos(u*Pi/2+Pi*3/2)-3*n,(2*n*sin(u*Pi/2+Pi*3/2)+2*n),t0],[n*cos(u*Pi/2+Pi*3/2)-3*n,-(2*n*sin(u*Pi/2+Pi*3/2)+2*n),t0]},u=0..1,style=line,axes=frame,scaling=constrained,lightmodel=light3,thickness=3,color=red):
c2:=rotate(c1,0,0,Pi/2):
c3:=rotate(c1,0,0,-Pi/2):
c4:=rotate(c1,0,0,Pi):
c5:=display([c1,c2,c3,c4]):
c6:=translate(c5,0,0,t1):
display([a,c1]):display([a,c5]):
display([a,c5,b,c6]):

```

```

>#interpolasi lengkung vertikal pola jaring
c11:=plot3d({[n*cos(u*Pi/2+Pi*3/2)-3*n, (2*n*sin(u*Pi/2+Pi*3/2)+2*n
),v*t0+(1-v)*t1],[n*cos(u*Pi/2+Pi*3/2)-3*n,-(2*n*sin(u*Pi/2+Pi*
3/2)+2*n),v*t0+(1-v)*t1]},v=0..1,u=0..1,style=patchnogrid,axes=
frame,scaling=constrained,lightmodel=light3,thickness=3):
c12:=rotate(c11,0,0,Pi/2):
c13:=rotate(c11,0,0,-Pi/2):
c14:=rotate(c11,0,0,Pi):
display([c11,c12,c13,c14]);

>#kurva pola fan pada grid persegi
d1:=spacecurve({[n*cos(u*Pi/2+Pi/2)-2*n, (2*n*sin(u*Pi/2+Pi/2)),t0]
,[n*cos(u*Pi/2+Pi*3/2)-3*n,-(2*n*sin(u*Pi/2+Pi*3/2)+2*n),t0]},u=0
..1,style=line,axes=frame,scaling=constrained,lightmodel=light3,thic
kness=3,color=red):
d2:=rotate(d1,0,0,Pi/2):
d3:=rotate(d1,0,0,-Pi/2):
d4:=rotate(d1,0,0,Pi):
d5:=display([d1,d2,d3,d4]):
d6:=translate(d5,0,0,t1):
display([a,d3]):
display([a,d5]):display([a,d5,b,d6]):

>#interpolasi lengkung vertikal pola fan
d11:=plot3d({[n*cos(u*Pi/2+Pi/2)-2*n, (2*n*sin(u*Pi/2+Pi/2)),v*t0+
(1-v)*t1],[n*cos(u*Pi/2+Pi*3/2)-3*n,-(2*n*sin(u*Pi/2+Pi*3/2)+2*n),
v*t0+(1-v)*t1]},v=0..1,u=0..1,style=patchnogrid,axes=frame,scaling
=constrained,lightmodel=light3,thickness=3):
d12:=rotate(d11,0,0,Pi/2):
d13:=rotate(d11,0,0,-Pi/2):
d14:=rotate(d11,0,0,Pi):
display([d11,d12,d13,d14]);

>#kurva pola keping pada grid persegi
e1:=spacecurve({[n*cos(u*Pi/2+Pi/2)-2*n, (n*sin(u*Pi/2+Pi/2)+n),t0]
,[n/2*cos(u*Pi/2)-3*n, (n*sin(u*Pi/2)+0*n),t0],[n*cos(u*Pi/2+Pi/2)-
2*n,-(n*sin(u*Pi/2+Pi/2)+n),t0],[n/2*cos(u*Pi/2)-3*n,-(n*sin(u*Pi
/2)+0*n),t0]},u=0..1,style=line,axes=frame,scaling=constrained,light
model=light3,thickness=3,color=red):
e2:=rotate(e1,0,0,Pi/2):
e3:=rotate(e1,0,0,-Pi/2):
e4:=rotate(e1,0,0,Pi):
e5:=display([e1,e2,e3,e4]):
e6:=translate(e5,0,0,t1):
display([a,e3]):
display([a,e5]):
display([a,e5,b,e6]):

>#interpolasi lengkung vertikal pola keping
e11:=plot3d({[n*cos(u*Pi/2+Pi/2)-2*n, (n*sin(u*Pi/2+Pi/2)+n),v*t0
+(1-v)*t1],[n/2*cos(u*Pi/2)-3*n, (n*sin(u*Pi/2)+0*n),v*t0+(1-v)*t1
],[n*cos(u*Pi/2+Pi/2)-2*n,-(n*sin(u*Pi/2+Pi/2)+n),v*t0+(1-v)*t1],
[n/2*cos(u*Pi/2)-3*n,-(n*sin(u*Pi/2)+0*n),v*t0+(1-v)*t1]},v=0..1
,u=0..1,style=patchnogrid,axes=frame,scaling=constrained,lightmodel=
light3,thickness=3):
e12:=rotate(e11,0,0,Pi/2):
e13:=rotate(e11,0,0,-Pi/2):

```



```
e14:=rotate(e11,0,0,Pi):
display([e11,e12,e13,e14]);
```

A.2 Pola Potongan Tegak Kasus Tongkat Cekung-cembung

```
> restart:
with (plots):
with (plottools):
panjang:=6; tinggi:=10: m:=4:
n:=tinggi/m: t0:=0:
t1:=tinggi:
p:=panjang/2: p1:=panjang/4:
c1:=p+p1:
c2:=p-p1:
c3:=p-p1:
c4:=p+p1:
c5:=p-p1-p1:
c6:=p-1/2*p1: k:=4:

> #kurva variasi kurva cembung p=9/2 dan cekung p=3/2
a1:=spacecurve([[(1-u)^2*(p)+2*(1-u)*(u)*(c1)+(u)^2*(p),0,(1-u)^2*(0*n)+2*(1-u)*(u)*(1*n/2)+(u)^2*(1*n)],[(1-u)^2*(p)+2*(1-u)*(u)*(c2)+(u)^2*(p),0,(1-u)^2*(1*n)+2*(1-u)*(u)*(3*n/2)+(u)^2*(2*n)],[(1-u)^2*(p)+2*(1-u)*(u)*(c3)+(u)^2*(p),0,(1-u)^2*(2*n)+2*(1-u)*(u)*(5*n/2)+(u)^2*(3*n)],[(1-u)^2*(p)+2*(1-u)*(u)*(c4)+(u)^2*(p),0,(1-u)^2*(3*n)+2*(1-u)*(u)*(7*n/2)+(u)^2*(4*n)]]],u=0..1,color=red,axes=frame,scaling=constrained,lightmodel=light3):
a2:=rotate(a1,0,0,Pi/2):
a3:=rotate(a1,0,0,Pi):
a4:=rotate(a1,0,0,-Pi/2):
display([a1]):

> plot3d([cos(2*Pi*v)*((1-u)^2*(p)+2*(1-u)*(u)*(c1)+(u)^2*(p)),sin(2*Pi*v)*((1-u)^2*(p)+2*(1-u)*(u)*(c1)+(u)^2*(p)),(1-u)^2*(0*n)+2*(1-u)*(u)*(1*n/2)+(u)^2*(1*n)], [cos(2*Pi*v)*((1-u)^2*(p)+2*(1-u)*(u)*(c2)+(u)^2*(p)),sin(2*Pi*v)*((1-u)^2*(p)+2*(1-u)*(u)*(c2)+(u)^2*(p)),(1-u)^2*(1*n)+2*(1-u)*(u)*(3*n/2)+(u)^2*(2*n)], [cos(2*Pi*v)*((1-u)^2*(p)+2*(1-u)*(u)*(c3)+(u)^2*(p)),sin(2*Pi*v)*((1-u)^2*(p)+2*(1-u)*(u)*(c3)+(u)^2*(p)),(1-u)^2*(2*n)+2*(1-u)*(u)*(5*n/2)+(u)^2*(3*n)], [cos(2*Pi*v)*((1-u)^2*(p)+2*(1-u)*(u)*(c4)+(u)^2*(p)),sin(2*Pi*v)*((1-u)^2*(p)+2*(1-u)*(u)*(c4)+(u)^2*(p)),(1-u)^2*(3*n)+2*(1-u)*(u)*(7*n/2)+(u)^2*(4*n)]]],v=0..1,u=0..1,style=patchnogrid,axes=frame,scaling=constrained,lightmodel=light3,labels=[x,y,z]):

> #kurva variasi kurva cembung p=9/2 dan cekung p=3/2
aa1:=spacecurve([[(1-u)^2*(p)+2*(1-u)*(u)*(c1)+(u)^2*(p),0,(1-u)^2*(0*n)+2*(1-u)*(u)*(1*n/2)+(u)^2*(1*n)],[(1-u)^2*(p)+2*(1-u)*(u)*(c4)+(u)^2*(p),0,(1-u)^2*(1*n)+2*(1-u)*(u)*(3*n/2)+(u)^2*(2*n)],[(1-u)^2*(p)+2*(1-u)*(u)*(c3)+(u)^2*(p),0,(1-u)^2*(2*n)+2*(1-u)*(u)*(5*n/2)+(u)^2*(3*n)],[(1-u)^2*(p)+2*(1-u)*(u)*(c4)+(u)^2*(p),0,(1-u)^2*(3*n)+2*(1-u)*(u)*(7*n/2)+(u)^2*(4*n)]]],u=0..1,color=red,axes=frame,scaling=constrained,lightmodel=light3):
aa2:=rotate(aa1,0,0,Pi/2):
aa3:=rotate(aa1,0,0,Pi):
aa4:=rotate(aa1,0,0,-Pi/2):
display([aa1]):

> leng1:=plot3d([cos(2*Pi*v)*((1-u)^2*(p)+2*(1-u)*(u)*(c1)+(u)^2*(p)),sin(2*Pi*v)*((1-u)^2*(p)+2*(1-u)*(u)*(c1)+(u)^2*(p)),(1-u)^2*(0*n)+2*(1-u)*(u)*(1*n/2)+(u)^2*(1*n)], [cos(2*Pi*v)*((1-u)^2*(p)+2*(1-u)*(u)*(c4)+(u)^2*(p)),sin(2*Pi*v)*((1-u)^2*(p)+2*(1-u)*(u)*(c4)+(u)^2*(p)),(1-u)^2*(1*n)+2*(1-u)*(u)*(3*n/2)+(u)^2*(2*n)], [cos(2*Pi*v)*((1-u)^2*(p)+2*(1-u)*(u)*(c3)+(u)^2*(p)),sin(2*Pi*v)*((1-u)^2*(p)+2*(1-u)*(u)*(c3)+(u)^2*(p)),(1-u)^2*(2*n)+2*(1-u)*(u)*(5*n/2)+(u)^2*(3*n)], [cos(2*Pi*v)*((1-u)^2*(p)+2*(1-u)*(u)*(c4)+(u)^2*(p)),sin(2*Pi*v)*((1-u)^2*(p)+2*(1-u)*(u)*(c4)+(u)^2*(p)),(1-u)^2*(3*n)+2*(1-u)*(u)*(7*n/2)+(u)^2*(4*n)]]],v=0..1,u=0..1,style=patchnogrid,axes=frame,scaling=constrained,lightmodel=light3,labels=[x,y,z]):
```



```
(0*n)+2*(1-u)*(u)*(1*n/2)+(u)^2*(1*n)], [cos(2*Pi*v)*((1-u)^2*(p)+2
*(1-u)*(u)*(c4)+(u)^2*(p)), sin(2*Pi*v)*((1-u)^2*(p)+2*(1-u)*(u)*
(c4)+(u)^2*(p)), (1-u)^2*(1*n)+2*(1-u)*(u)*(3*n/2)+(u)^2*(2*n)], [
cos(2*Pi*v)*((1-u)^2*(p)+2*(1-u)*(u)*(c3)+(u)^2*(p)), sin(2*Pi*v)*
(1-u)^2*(p)+2*(1-u)*(u)*(c3)+(u)^2*(p)), (1-u)^2*(2*n)+2*(1-u)*(u)
*(5*n/2)+(u)^2*(3*n)], [cos(2*Pi*v)*((1-u)^2*(p)+2*(1-u)*(u)*(c4)+
(u)^2*(p)), sin(2*Pi*v)*((1-u)^2*(p)+2*(1-u)*(u)*(c4)+(u)^2*(p)), (1
-u)^2*(3*n)+2*(1-u)*(u)*(7*n/2)+(u)^2*(4*n)]]; v=0..1, u=0..1, style=
patchnogrid, axes=frame, scaling=constrained, lightmodel=light3, labels=
[x, y, z]):
display([leng1, s]);
```

```
>#kurva variasi kurva cembung p=9/2 dan cekung p=9/4
aaa1:=spacecurve({[(1-u)^2*(p)+2*(1-u)*(u)*(c1)+(u)^2*(p), 0, (1-u)
^2*(0*n)+2*(1-u)*(u)*(1*n/2)+(u)^2*(1*n)], [(1-u)^2*(p)+2*(1-u)*(u)
*(c2)+(u)^2*(p), 0, (1-u)^2*(1*n)+2*(1-u)*(u)*(3*n/2)+(u)^2*(2*n)]
, [(1-u)^2*(p)+2*(1-u)*(u)*(c4)+(u)^2*(p), 0, (1-u)^2*(2*n)+2*(1-u)
*(u)*(5*n/2)+(u)^2*(3*n)], [(1-u)^2*(p)+2*(1-u)*(u)*(c3)+(u)^2*(p),
0, (1-u)^2*(3*n)+2*(1-u)*(u)*(7*n/2)+(u)^2*(4*n)]}, u=0..1, color=
red, axes=frame, scaling=constrained, lightmodel=light3):
aaa2:=rotate(a1, 0, 0, Pi/2):
aaa3:=rotate(a1, 0, 0, Pi):
aaa4:=rotate(a1, 0, 0, -Pi/2):
display([aaa1]):
>plot3d({[cos(2*Pi*v)*((1-u)^2*(p)+2*(1-u)*(u)*(c1)+(u)^2*(p)), sin
(2*Pi*v)*((1-u)^2*(p)+2*(1-u)*(u)*(c1)+(u)^2*(p)), (1-u)^2*(0*n)+2*
(1-u)*(u)*(1*n/2)+(u)^2*(1*n)], [cos(2*Pi*v)*((1-u)^2*(p)+2*(1-u)*
(u)*(c1)+(u)^2*(p)), sin(2*Pi*v)*((1-u)^2*(p)+2*(1-u)*(u)*(c1)+(u)^
2*(p)), (1-u)^2*(1*n)+2*(1-u)*(u)*(3*n/2)+(u)^2*(2*n)], [cos(2*Pi*v)
*((1-u)^2*(p)+2*(1-u)*(u)*(c4)+(u)^2*(p)), sin(2*Pi*v)*((1-u)^2*(p)
+2*(1-u)*(u)*(c4)+(u)^2*(p)), (1-u)^2*(2*n)+2*(1-u)*(u)*(5*n/2)+(u)
^2*(3*n)], [cos(2*Pi*v)*((1-u)^2*(p)+2*(1-u)*(u)*(c4)+(u)^2*(p)),
sin(2*Pi*v)*((1-u)^2*(p)+2*(1-u)*(u)*(c4)+(u)^2*(p)), (1-u)^2*(3*n)
+2*(1-u)*(u)*(7*n/2)+(u)^2*(4*n)]}, v=0..1, u=0..1, style=patchnogrid
, axes=frame, scaling=constrained, lightmodel=light3, labels=[x, y, z]):
```

```
>#kurva variasi kurva cembung p=9/2
a:=spacecurve({[(1-u)^2*(p)+2*(1-u)*(u)*(c1)+(u)^2*(p), 0, (1-u)^2*
(0*n)+2*(1-u)*(u)*(1*n/2)+(u)^2*(1*n)]}, u=0..1, color=red, axes=fra
me, scaling=constrained, lightmodel=light3):
a2:=translate(a, 0, 0, 5):
>aa:=plot3d({[cos(2*Pi*v)*((1-u)^2*(p)+2*(1-u)*(u)*(c1)+(u)^2*(p))
, sin(2*Pi*v)*((1-u)^2*(p)+2*(1-u)*(u)*(c1)+(u)^2*(p)), (1-u)^2*(0*
n)+2*(1-u)*(u)*(1*n/2)+(u)^2*(1*n)]}, v=0..1, u=0..1, style=patchno
grid, axes=frame, scaling=constrained, lightmodel=light3):
aaa:=translate(aa, 0, 0, 5):
```

A.3 Pola Potongan Tegak Kasus Tongkat Kepingan

```
> restart:
with (plots):
with (plottools):
panjang:=10; tinggi:=32: m:=4:
n:=tinggi/m: t0:=0:
t1:=tinggi:
n1:=n/8:
p:=panjang/10:
```

k:=4:

```
> #grid persegi panjang dari potongan tegak
grs1:=spacecurve([0,0,t],t=0..8,color=blue):
grs2:=translate(grs1,1/8*n,0,0):
grs3:=translate(grs1,2/8*n,0,0):
grs4:=translate(grs1,3/8*n,0,0):
grs5:=translate(grs1,4/8*n,0,0):
grs6:=translate(grs1,5/8*n,0,0):
grs11:=spacecurve([1,0,0],l=0..5,color=blue):
grs12:=translate(grs11,0,0,1/8*n):
grs13:=translate(grs11,0,0,2/8*n):
grs14:=translate(grs11,0,0,3/8*n):
grs15:=translate(grs11,0,0,4/8*n):
grs16:=translate(grs11,0,0,5/8*n):
grs17:=translate(grs11,0,0,6/8*n):
grs18:=translate(grs11,0,0,7/8*n):
grs19:=translate(grs11,0,0,8/8*n):
a:=display([grs1,grs2,grs3,grs4,grs5,grs6,grs11,grs12,grs13,grs14,grs15,grs16,grs17,grs18,grs19]):
display([a],style=line,axes=frame,scaling=constrained,lightmodel=light3):
c:=translate(a,0,0,8):

> #kasus kepingan kurva pola zigzag
a1:=spacecurve({[(p*cos(u*Pi/2)+2*p)+n/4,0,(p*sin(u*Pi/2)+0)],[(p*cos(u*Pi/2)+1*p)+n/4,0,(p*sin(u*Pi/2)+n1)],[(p*cos(-u*Pi/2)+1*p)+n/4,0,(p*sin(-u*Pi/2)+3*n1)],[(p*cos(-u*Pi/2)+2*p)+n/4,0,(p*sin(-u*Pi/2)+4*n1)],[(p*cos(u*Pi/2)+2*p)+n/4,0,(p*sin(u*Pi/2)+4*n1)],[(p*cos(u*Pi/2)+1*p)+n/4,0,(p*sin(u*Pi/2)+5*n1)],[(p*cos(-u*Pi/2)+1*p)+n/4,0,(p*sin(-u*Pi/2)+7*n1)],[(p*cos(-u*Pi/2)+2*p)+n/4,0,(p*sin(-u*Pi/2)+8*n1)]},u=0..1,style=line,axes=frame,scaling=constrained,lightmodel=light3,thickness=3,color=red):
a2:=translate(a1,0,0,n):
a3:=translate(a1,0,0,2*n):
a4:=translate(a1,0,0,3*n):
display([a,a1],labels=[x,y,z]):
display([a,c,a1,a2],labels=[x,y,z]):
>
a11:=plot3d({[cos(v*2*Pi)*((p*cos(u*Pi/2)+2*p)+n/4),sin(v*2*Pi)*((p*cos(u*Pi/2)+2*p)+n/4),(p*sin(u*Pi/2)+0)],cos(v*2*Pi)*((p*cos(-u*Pi/2)+1*p)+n/4),sin(v*2*Pi)*((p*cos(-u*Pi/2)+1*p)+n/4),(p*sin(-u*Pi/2)+3*n1)],cos(v*2*Pi)*((p*cos(-u*Pi/2)+2*p)+n/4),sin(v*2*Pi)*((p*cos(-u*Pi/2)+2*p)+n/4),(p*sin(-u*Pi/2)+4*n1)],cos(v*2*Pi)*((p*cos(u*Pi/2)+1*p)+n/4),sin(v*2*Pi)*((p*cos(u*Pi/2)+1*p)+n/4),(p*sin(u*Pi/2)+n1)],cos(v*2*Pi)*((p*cos(u*Pi/2)+2*p)+n/4),sin(v*2*Pi)*((p*cos(u*Pi/2)+2*p)+n/4),(p*sin(u*Pi/2)+4*n1)],cos(v*2*Pi)*((p*cos(u*Pi/2)+1*p)+n/4),sin(v*2*Pi)*((p*cos(u*Pi/2)+1*p)+n/4),(p*sin(u*Pi/2)+5*n1)],cos(v*2*Pi)*((p*cos(-u*Pi/2)+1*p)+n/4),sin(v*2*Pi)*((p*cos(-u*Pi/2)+1*p)+n/4),(p*sin(-u*Pi/2)+7*n1)],cos(v*2*Pi)*((p*cos(-u*Pi/2)+2*p)+n/4),sin(v*2*Pi)*((p*cos(-u*Pi/2)+2*p)+n/4),(p*sin(-u*Pi/2)+8*n1)]},v=0..1,u=0..1,style=patchnogrid,axes=normal,scaling=constrained,lightmodel=light3,thickness=3):
a12:=translate(a11,0,0,n):
a13:=translate(a11,0,0,2*n):
a14:=translate(a11,0,0,3*n):
display(a11):
```

```

display([a11,a12,s],labels=[x,y,z]);

>#kamus kepingan kurva pola miring
b1:=spacecurve({[(p*sqrt(2)*cos(u*Pi/2+Pi*3/4)+2*p)+n/4],0,(p*sqrt(2)*sin(u*Pi/2+Pi*3/4)+n1)],[(p*cos(-u*Pi/2)+1*p)+n/4],0,(p*2*sin(-u*Pi/2)+4*n1)],[(p*sqrt(2)*cos(u*Pi/2+Pi*3/4)+3*p)+n/4],0,(p*sqrt(2)*sin(u*Pi/2+Pi*3/4)+5*n1)],[(p*cos(-u*Pi/2)+2*p)+n/4],0,(p*2*sin(-u*Pi/2)+8*n1)]},u=0..1,style=line,axes=frame,scaling=constrained,lightmodel=light3,thickness=3,color=red):
b2:=spacecurve({[(p*sqrt(2)*cos(u*Pi/2+Pi*3/4)+2*p),0,-(p*sqrt(2)*sin(u*Pi/2+Pi*3/4)+n1)+2*n],[(p*cos(-u*Pi/2)+1*p),0,-(p*2*sin(-u*Pi/2)+4*n1)+2*n],[(p*sqrt(2)*cos(u*Pi/2+Pi*3/4)+3*p),0,-(p*sqrt(2)*sin(u*Pi/2+Pi*3/4)+5*n1)+2*n],[(p*cos(-u*Pi/2)+2*p),0,-(p*2*sin(-u*Pi/2)+8*n1)+2*n]}},u=0..1,style=line,axes=frame,scaling=constrained,lightmodel=light3,thickness=3,color=red):
b3:=display([b1,b2]):
b4:=translate(b3,0,0,2*n):
display([a,b1]):
>
b11:=plot3d({[cos(v*2*Pi)*((p*sqrt(2)*cos(u*Pi/2+Pi*3/2)+2*p)+n/6.5),sin(v*2*Pi)*((p*sqrt(2)*cos(u*Pi/2+Pi*3/2)+2*p)+n/6.5),(p*sqrt(2)*sin(u*Pi/2+Pi*3/4)+n1)],[(cos(v*2*Pi)*((p*cos(-u*Pi/2)+1*p)+n/4),sin(v*2*Pi)*((p*cos(-u*Pi/2)+1*p)+n/4),(p*2*sin(-u*Pi/2)+4*n1)]},[cos(v*2*Pi)*((p*sqrt(2)*cos(u*Pi/2+Pi*3/4)+3*p)+n/4),sin(v*2*Pi)*((p*sqrt(2)*cos(u*Pi/2+Pi*3/4)+3*p)+n/4),(p*sqrt(2)*sin(u*Pi/2+Pi*3/4)+5*n1)],[(cos(v*2*Pi)*((p*cos(-u*Pi/2)+2*p)+n/4),sin(v*2*Pi)*((p*cos(-u*Pi/2)+2*p)+n/4),(p*2*sin(-u*Pi/2)+8*n1)]},v=0..1,u=0..1,style=patchnogrid,axes=normal,scaling=constrained,lightmodel=light3,thickness=3):
b12:=plot3d({[cos(v*2*Pi)*((p*sqrt(2)*cos(u*Pi/2+Pi*3/2)+2*p)+n/6.5),sin(v*2*Pi)*((p*sqrt(2)*cos(u*Pi/2+Pi*3/2)+2*p)+n/6.5),(p*sqrt(2)*sin(u*Pi/2+Pi*3/4)+n1)+2*n],[(cos(v*2*Pi)*((p*cos(-u*Pi/2)+1*p)+n/4),sin(v*2*Pi)*((p*cos(-u*Pi/2)+1*p)+n/4),-(p*2*sin(-u*Pi/2)+4*n1)+2*n],[(cos(v*2*Pi)*((p*sqrt(2)*cos(u*Pi/2+Pi*3/4)+3*p)+n/4),sin(v*2*Pi)*((p*sqrt(2)*cos(u*Pi/2+Pi*3/4)+3*p)+n/4),-(p*sqrt(2)*sin(u*Pi/2+Pi*3/4)+5*n1)+2*n],[(cos(v*2*Pi)*((p*cos(-u*Pi/2)+2*p)+n/4),sin(v*2*Pi)*((p*cos(-u*Pi/2)+2*p)+n/4),-(p*2*sin(-u*Pi/2)+8*n1)+2*n]}},v=0..1,u=0..1,style=patchnogrid,axes=normal,scaling=constrained,lightmodel=light3,thickness=3):
b13:=display([b11,b12]):
b14:=translate(b11,0,0,2*n):
display([b11,b12]);

>#kamus kepingan kurva pola simetris
c1:=spacecurve({[(p*cos(u*Pi/2)+2*p)+n/4],0,(n1*sin(u*Pi/2)+0)],[(p*cos(u*Pi/2+Pi)+2*p)+n/4],0,(n1*sin(u*Pi/2+Pi)+2*n1)],[(p/4*cos(u*Pi-Pi/2)+p)+n/4],0,(n1/2*sin(u*Pi-Pi/2)+5/2*n1)],[(p*cos(u*Pi/2-Pi/2)+p)+n/4],0,(n1*sin(u*Pi/2-Pi/2)+4*n1)]},u=0..1,style=line,axes=frame,scaling=constrained,lightmodel=light3,thickness=3,color=red):
c2:=spacecurve({[(p*cos(u*Pi/2)+2*p)+n/4],0,-(n1*sin(u*Pi/2)+0)+n],[(p*cos(u*Pi/2+Pi)+2*p)+n/4],0,-(n1*sin(u*Pi/2+Pi)+2*n1)+n],[(p/4*cos(u*Pi-Pi/2)+p)+n/4],0,-(n1/2*sin(u*Pi-Pi/2)+5/2*n1)+n],[(p*cos(u*Pi/2-Pi/2)+p)+n/4],0,-(n1*sin(u*Pi/2-Pi/2)+4*n1)+n]}},u=0..1,style=line,axes=frame,scaling=constrained,lightmodel=light3,thickness=3,color=red):
c3:=display([c1,c2]):

```

```

c4:=translate(c3,0,0,1*n):
c5:=translate(c3,0,0,2*n):
c6:=translate(c3,0,0,3*n):
display([a,c3]):
display([c3,c4,c5]):
>
c11:=plot3d({[cos(v*2*Pi)*(p*cos(u*Pi/2)+3*p/2)+n/4],sin(v*2*Pi)*((p*cos(u*Pi/2)+3*p/2)+n/4),(n1*sin(u*Pi/2)+0)],
[cos(v*2*Pi)*(p*cos(u*Pi/2+Pi)+2*p)+n/4],sin(v*2*Pi)*((p*cos(u*Pi/2+Pi)+2*p)+n/4),(n1*sin(u*Pi/2+Pi)+2*n1)],
[cos(v*2*Pi)*(p/4*cos(u*Pi-Pi/2)+p)+n/4],sin(v*2*Pi)*((p/4*cos(u*Pi-Pi/2)+p)+n/4),(n1/2*sin(u*Pi-Pi/2)+5/2*n1)],
[cos(v*2*Pi)*(p*cos(u*Pi/2-Pi/2)+p)+n/4],sin(v*2*Pi)*((p*cos(u*Pi/2-Pi/2)+p)+n/4),(n1*sin(u*Pi/2-Pi/2)+4*n1)]},v=0..1,u=0..1,
style=patchnogrid,axes=normal,scaling=constrained,lightmodel=light3,thickness=3):
c12:=plot3d({[cos(v*2*Pi)*(p*cos(u*Pi/2)+3*p/2)+n/4],sin(v*2*Pi)*((p*cos(u*Pi/2)+3*p/2)+n/4),-(n1*sin(u*Pi/2)+0)+n],
[cos(v*2*Pi)*((p*cos(u*Pi/2+Pi)+2*p)+n/4),sin(v*2*Pi)*((p*cos(u*Pi/2+Pi)+2*p)+n/4),-(n1*sin(u*Pi/2+Pi)+2*n1)+n],
[cos(v*2*Pi)*((p/4*cos(u*Pi-Pi/2)+p)+n/4),-(n1/2*sin(u*Pi-Pi/2)+5/2*n1)+n],
[cos(v*2*Pi)*((p*cos(u*Pi/2-Pi/2)+p)+n/4),sin(v*2*Pi)*((p*cos(u*Pi/2-Pi/2)+p)+n/4),-(n1*sin(u*Pi/2-Pi/2)+4*n1)+n]}},v=0..1,u=0..1,
style=patchnogrid,axes=normal,scaling=constrained,lightmodel=light3,thickness=3):
c13:=display([c11,c12]):
c14:=translate(c13,0,0,1*n):
c15:=translate(c13,0,0,2*n):
c16:=translate(c13,0,0,3*n):
display([c13]):display([c13,c14]):
>#kasus kepingan kurva pola campuran
d1:=spacecurve({[(p*cos(u*Pi/2)+2*p)+n/4],0,(n1*sin(u*Pi/2)+0)],
[(p/2*cos(u*Pi-Pi/2)+2*p)+n/4],0,(n1*sin(u*Pi-Pi/2)+2*n1)],
[(p*cos(u*Pi/2+Pi)+2*p)+n/4],0,(n1*sin(u*Pi/2+Pi)+4*n1)],
[(p*cos(u*Pi/2+Pi/2)+2*p)+n/4],0,(n1*sin(u*Pi/2+Pi/2)+4*n1)],
[(p*cos(u*Pi-Pi/2)+2*p)+n/4],0,(n1*sin(u*Pi-Pi/2)+6*n1)],
[(p*cos(u*Pi/2)+p)+n/4],0,(n1*sin(u*Pi/2)+7*n1)]},u=0..1,style=line,axes=frame,scaling=constrained,lightmodel=light3,thickness=3,color=red):
d2:=spacecurve({[(p*cos(u*Pi/2)+2*p)+n/4],0,-(n1*sin(u*Pi/2)+0)+2*n],
[(p/2*cos(u*Pi-Pi/2)+2*p)+n/4],0,-(n1*sin(u*Pi-Pi/2)+2*n1)+2*n],
[(p*cos(u*Pi/2+Pi)+2*p)+n/4],0,-(n1*sin(u*Pi/2+Pi)+4*n1)+2*n],
[(p*cos(u*Pi/2+Pi/2)+2*p)+n/4],0,-(n1*sin(u*Pi/2+Pi/2)+4*n1)+2*n],
[(p*cos(u*Pi-Pi/2)+2*p)+n/4],0,-(n1*sin(u*Pi-Pi/2)+6*n1)+2*n],
[(p*cos(u*Pi/2)+p)+n/4],0,-(n1*sin(u*Pi/2)+7*n1)+2*n]}},u=0..1,style=line,axes=frame,scaling=constrained,lightmodel=light3,thickness=3,color=red):
d3:=display([d1,d2]):
d4:=translate(d3,0,0,2*n):
display([a,d1]):
display([d3,d4]):
>
d11:=plot3d({[cos(v*2*Pi)*(p*cos(u*Pi/2)+2*p)+n/4],sin(v*2*Pi)*((p*cos(u*Pi/2)+2*p)+n/4),(n1*sin(u*Pi/2)+0)],
[cos(v*2*Pi)*((p/2*cos(u*Pi-Pi/2)+2*p)+n/4),sin(v*2*Pi)*((p/2*cos(u*Pi-Pi/2)+2*p)+n/4),
(n1*sin(u*Pi-Pi/2)+2*n1)],
[cos(v*2*Pi)*((p*sqrt(2)*cos(u*Pi/2+Pi*3/4)+3*p)+n/4),sin(v*2*Pi)*((p*sqrt(2)*cos(u*Pi/2+Pi*3/4)+3*p)+n/4),
-(p*sqrt(2)*sin(u*Pi/2+Pi*3/4)+5*n1)+5.63/5*n],
[cos(v*2*Pi)*((p*

```



```

cos(u*Pi-Pi/2)+2*p)+n/4), sin(v*2*Pi) * ((p*cos(u*Pi-Pi/2)+2*p)+n/4)
, (n1*sin(u*Pi-Pi/2)+6*n1)], [cos(v*2*Pi) * ((p*cos(u*Pi/2)+p)+n/4),
sin(v*2*Pi) * ((p*cos(u*Pi/2)+p)+n/4), (n1*sin(u*Pi/2)+7*n1)]}, v=0..1, u
=0..1, style=patchnogrid, axes=normal, scaling=constrained, lightmodel=1
ight3, thickness=3):
d12:=plot3d({ [cos(v*2*Pi) * ((p*cos(u*Pi/2)+2*p)+n/4), sin(v*2*Pi) * ((p*
cos(u*Pi/2)+2*p)+n/4), -(n1*sin(u*Pi/2)+0)+2*n], [cos(v*2*Pi) * ((
p/2*cos(u*Pi-Pi/2)+2*p)+n/4), sin(v*2*Pi) * ((p/2*cos(u*Pi-Pi/2)+2*p)
+n/4), -(n1*sin(u*Pi-Pi/2)+2*n1)+2*n], [cos(v*2*Pi) * ((p*sqrt(2)*cos
(u*Pi/2+Pi*3/4)+3*p)+n/4), sin(v*2*Pi) * ((p*sqrt(2)*cos(u*Pi/2+Pi*3/4)
+3*p)+n/4), -(p*sqrt(2)*sin(u*Pi/2+Pi*3/4)+5*n1)+2.125*n], [cos(v*
2*Pi) * ((p*cos(u*Pi-Pi/2)+2*p)+n/4), sin(v*2*Pi) * ((p*cos(u*Pi-Pi/2)
+2*p)+n/4), -(n1*sin(u*Pi-Pi/2)+6*n1)+2*n], [cos(v*2*Pi) * ((p*cos(u*
Pi/2)+p)+n/4), sin(v*2*Pi) * ((p*cos(u*Pi/2)+p)+n/4), -(n1*sin(u*Pi/2)
+7*n1)+2*n]}}, v=0..1, u=0..1, style=patchnogrid, axes=normal, scaling=con
strained, lightmodel=light3, thickness=3):
d13:=display([d11, d12]):
d14:=translate(d13, 0, 0, 2*n):
display([d13]):
display([d13]):

```

A.4 Pola Tongkat Puntiran

> #puntiran pola bunga

```

a1:=spacecurve({ [n*sqrt(2)*cos(u*Pi/2+Pi*3/4)-
n, n*sqrt(2)*sin(u*Pi/2+Pi*3/4)+n, t0], [n*sqrt(2)*cos(u*Pi/2+Pi*3/4)-
n, (n*sqrt(2)*sin(u*Pi/2+Pi*3/4)-
n), t0]}}, u=0..1, style=line, axes=frame, scaling=constrained, lightmodel=
light3, thickness=3, color=red):
a2:=rotate(a1, 0, 0, Pi/2):
a3:=rotate(a1, 0, 0, -Pi/2):
a4:=rotate(a1, 0, 0, Pi):
a5:=display([a1, a2, a3, a4]):
a21:=spacecurve({ [cos(sud) * (n*sqrt(2)*cos(u*Pi/2+Pi*3/4)-n) -sin
(sud) * (n*sqrt(2)*sin(u*Pi/2+Pi*3/4)+n), sin(sud) * (n*sqrt(2)*cos(u*Pi/
2+Pi*3/4)-n) +cos(sud) * (n*sqrt(2)*sin(u*Pi/2+Pi*3/4)+n), t1], [cos
(sud) * (n*sqrt(2)*cos(u*Pi/2+Pi*3/4)-n) -sin(sud) * (n*sqrt(2)*sin(u*
Pi/2+Pi*3/4)-n), sin(sud) * (n*sqrt(2)*cos(u*Pi/2+Pi*3/4)-n) +cos(sud)
* (n*sqrt(2)*sin(u*Pi/2+Pi*3/4)-n), t1]}}, u=0..1, style=line, axes=f
rame, scaling=constrained, lightmodel=light3, thickness=3, color=red):
a22:=rotate(a21, 0, 0, Pi/2):
a23:=rotate(a21, 0, 0, -Pi/2):
a24:=rotate(a21, 0, 0, Pi):
a6:=display([a21, a22, a23, a24]):
display([a, a1]):
display([a, a5]):
display([a, b, a5, a6], labels=[x, y, z]):
>a11:=plot3d({ [(v) * (n*sqrt(2)*cos(u*Pi/2+Pi*3/4)-n) + (1-v) * (cos(sud)
) * (n*sqrt(2)*cos(u*Pi/2+Pi*3/4)-n) -sin(sud) * (n*sqrt(2)*sin(u*Pi/2
+Pi*3/4)+n)), (v) * (n*sqrt(2)*sin(u*Pi/2+Pi*3/4)+n) + (1-v) * (sin(sud)
* (n*sqrt(2)*cos(u*Pi/2+Pi*3/4)-n) +cos(sud) * (n*sqrt(2)*sin(u*Pi/2+
Pi*3/4)+n)), (v) * (t0) + (1-v) * (t1)], [(v) * (n*sqrt(2)*cos(u*Pi/2+Pi*3/
4)-n) + (1-v) * (cos(sud) * (n*sqrt(2)*cos(u*Pi/2+Pi*3/4)-n) -sin(sud) *
(n*sqrt(2)*sin(u*Pi/2+Pi*3/4)-n)), (v) * ((n*sqrt(2)*sin(u*Pi/2+Pi*3/
4)-n) + (1-v) * (sin(sud) * (n*sqrt(2)*cos(u*Pi/2+Pi*3/4)-n) +cos(sud)
* (n*sqrt(2)*sin(u*Pi/2+Pi*3/4)-n)), (v) * (t0) + (1-v) * (t1)]}, v=0..1, u=

```

```

0..1,style=patchngrid,axes=normal,scaling=constrained,lightmodel=light3,thickness=3):
a12:=rotate(a11,0,0,Pi/2):
a13:=rotate(a11,0,0,-Pi/2):
a14:=rotate(a11,0,0,Pi):
display([a11,a12,a13,a14],labels=[x,y,z]):

>#puntiran pola sayap
b1:=spacecurve({[n*cos(u*Pi/2+Pi*2/4)-2*n,n*sin(u*Pi/2+Pi*2/4)+n,
t0],[2*n*cos(u*Pi/2+Pi)-n,(n*sin(u*Pi/2+Pi)+n),t0],[n*cos(u*Pi/2+
Pi*2/4)-2*n,-(n*sin(u*Pi/2+Pi*2/4)+n),t0],[2*n*cos(u*Pi/2+Pi)-n,-
((n*sin(u*Pi/2+Pi)+n)),t0]},u=0..1,style=line,axes=frame,scaling=con
strained,lightmodel=light3,thickness=3,color=red):
b2:=rotate(b1,0,0,Pi/2):
b3:=rotate(b1,0,0,-Pi/2):
b4:=rotate(b1,0,0,Pi):
b5:=display([b1,b2,b3,b4]):
b21:=spacecurve({[cos(sud)*(n*cos(u*Pi/2+Pi*2/4)-2*n)-sin(sud)*(n*
sin(u*Pi/2+Pi*2/4)+n),sin(sud)*(n*cos(u*Pi/2+Pi*2/4)-2*n)+cos(sud)
*(n*sin(u*Pi/2+Pi*2/4)+n),t1],[cos(sud)*(2*n*cos(u*Pi/2+Pi)-n)-
sin(sud)*(n*sin(u*Pi/2+Pi)+n),sin(sud)*(2*n*cos(u*Pi/2+Pi)-n)+cos
(sud)*(n*sin(u*Pi/2+Pi)+n),t1],[cos(sud)*(n*cos(u*Pi/2+Pi*2/4)-2*n)-
sin(sud)*(-(n*sin(u*Pi/2+Pi*2/4)+n)),sin(sud)*(n*cos(u*Pi/2
+Pi*2/4)-2*n)+cos(sud)*(-(n*sin(u*Pi/2+Pi*2/4)+n)),t1],[cos(sud)
*(2*n*cos(u*Pi/2+Pi)-n)-sin(sud)*(-(n*sin(u*Pi/2+Pi)+n)),sin
(sud)*(2*n*cos(u*Pi/2+Pi)-n)+cos(sud)*(-(n*sin(u*Pi/2+Pi)+n)),t1
]},u=0..1,style=line,axes=frame,scaling=constrained,lightmodel=light
3,thickness=3,color=red):
b22:=rotate(b21,0,0,Pi/2):
b23:=rotate(b21,0,0,-Pi/2):
b24:=rotate(b21,0,0,Pi):
b6:=display([b21,b22,b23,b24]):
display([a,b1]):
display([a,b5]):
display([a,b5,b,b6],labels=[x,y,z]):
>b11:=plot3d({[v*(n*cos(u*Pi/2+Pi*2/4)-2*n)+(1-v)*(cos(sud)*(n*cos
(u*Pi/2+Pi*2/4)-2*n)-sin(sud)*(n*sin(u*Pi/2+Pi*2/4)+n)),v*(n*sin
(u*Pi/2+Pi*2/4)+n)+(1-v)*(sin(sud)*(n*cos(u*Pi/2+Pi*2/4)-2*n)+cos
(sud)*(n*sin(u*Pi/2+Pi*2/4)+n)),v*t0+(1-v)*t1],[v*(2*n*cos(u*Pi/2
+Pi)-n)+(1-v)*(cos(sud)*(2*n*cos(u*Pi/2+Pi)-n)-sin(sud)*(n*sin
(u*Pi/2+Pi)+n)),v*(n*sin(u*Pi/2+Pi)+n)+(1-v)*(sin(sud)*(2*n*cos
(u*Pi/2+Pi)-n)+cos(sud)*(n*sin(u*Pi/2+Pi)+n)),v*t0+(1-v)*t1],[v*
(n*cos(u*Pi/2+Pi*2/4)-2*n)+(1-v)*(cos(sud)*(n*cos(u*Pi/2+Pi*2/4)-
2*n)-sin(sud)*(-(n*sin(u*Pi/2+Pi*2/4)+n))),v*(-(n*sin(u*Pi/2+Pi*
2/4)+n)+(1-v)*(sin(sud)*(n*cos(u*Pi/2+Pi*2/4)-2*n)+cos(sud)*(-
(n*sin(u*Pi/2+Pi*2/4)+n))),v*t0+(1-v)*t1],[v*(2*n*cos(u*Pi/2+Pi)-
n)+(1-v)*(cos(sud)*(2*n*cos(u*Pi/2+Pi)-n)-sin(sud)*(-(n*sin
(u*Pi/2+Pi)+n))),v*(-(n*sin(u*Pi/2+Pi)+n))+1-v*(sin(sud)*(2*n
*cos(u*Pi/2+Pi)-n)+cos(sud)*(-(n*sin(u*Pi/2+Pi)+n))),v*t0+(1-
v)*t1]},v=0..1,u=0..1,style=patchngrid,axes=normal,scaling=constrai
ned,lightmodel=light3,thickness=3):
b12:=rotate(b11,0,0,Pi/2):
b13:=rotate(b11,0,0,-Pi/2):
b14:=rotate(b11,0,0,Pi):
display([b11,b12,b13,b14,f],labels=[x,y,z]);

>#puntiran pola jaring

```



```

c1:=spacecurve({[n*cos(u*Pi/2+Pi*3/2)-3*n,(2*n*sin(u*Pi/2+Pi*3/2)
+2*n),t0],[n*cos(u*Pi/2+Pi*3/2)-3*n,-(2*n*sin(u*Pi/2+Pi*3/2)+2*n)
,t0]},u=0..1,style=line,axes=frame,scaling=constrained,lightmodel=li
ght3,thickness=3,color=red):
c2:=rotate(c1,0,0,Pi/2):
c3:=rotate(c1,0,0,-Pi/2):
c4:=rotate(c1,0,0,pi):
c5:=display([c1,c2,c3,c4]):
c21:=spacecurve({[cos(sud)*(n*cos(u*Pi/2+Pi*3/2)-3*n)-sin(sud)*(
2*n*sin(u*Pi/2+Pi*3/2)+2*n),sin(sud)*(n*cos(u*Pi/2+Pi*3/2)-3*n)
+cos(sud)*(2*n*sin(u*Pi/2+Pi*3/2)+2*n),t1],[cos(sud)*(n*cos(u*Pi/2+P
i*3/2)-3*n)-sin(sud)*(-(2*n*sin(u*Pi/2+Pi*3/2)+2*n)),sin(sud)*(n
*cos(u*Pi/2+Pi*3/2)-3*n)+cos(sud)*(-(2*n*sin(u*Pi/2+Pi*3/2)+2*n)
),t1]},u=0..1,style=line,axes=frame,scaling=constrained,lightmodel=li
ght3,thickness=3,color=red):
c22:=rotate(c21,0,0,pi/2):
c23:=rotate(c21,0,0,-pi/2):
c24:=rotate(c21,0,0,pi):
c6:=display([c21,c22,c23,c24]):
display([a,c1]):
display([a,c5]):
display([a,c5,b,c6]):
>c11:=plot3d({[v*(n*cos(u*Pi/2+Pi*3/2)-3*n)+(1-v)*(cos(sud)*(n*cos
(u*Pi/2+Pi*3/2)-3*n)-sin(sud)*(2*n*sin(u*Pi/2+Pi*3/2)+2*n)),v*(2*n
*sin(u*Pi/2+Pi*3/2)+2*n)+(1-v)*(sin(sud)*(n*cos(u*Pi/2+Pi*3/2)-3*
n)+cos(sud)*(2*n*sin(u*Pi/2+Pi*3/2)+2*n)),v*t0+(1-v)*t1],[v*(n*cos
(u*Pi/2+Pi*3/2)-3*n)+(1-v)*(cos(sud)*(n*cos(u*Pi/2+Pi*3/2)-3*n)-
sin(sud)*(-(2*n*sin(u*Pi/2+Pi*3/2)+2*n))),v*(-(2*n*sin(u*Pi/2+Pi*
3/2)+2*n)+(1-v)*(sin(sud)*(n*cos(u*Pi/2+Pi*3/2)-3*n)+cos(sud)*(-
(2*n*sin(u*Pi/2+Pi*3/2)+2*n))),v*t0+(1-v)*t1]},v=0..1,u=0..1,
style=patchnogrid,axes=frame,scaling=constrained,lightmodel=light3,t
hickness=3,color=red):
c12:=rotate(c11,0,0,pi/2):
c13:=rotate(c11,0,0,-pi/2):
c14:=rotate(c11,0,0,pi):
display([c11,c12,c13,c14]);

>#puntiran pola fan
d1:=spacecurve({[n*cos(u*Pi/2+Pi/2)-2*n,(2*n*sin(u*Pi/2+Pi/2)),t0]
,[n*cos(u*Pi/2+Pi*3/2)-3*n,-(2*n*sin(u*Pi/2+Pi*3/2)+2*n),t0]},u=0
..1,style=line,axes=frame,scaling=constrained,lightmodel=light3,thic
kness=3,color=red):
d2:=rotate(d1,0,0,pi/2):
d3:=rotate(d1,0,0,-pi/2):
d4:=rotate(d1,0,0,pi):
d5:=display([d1,d2,d3,d4]):
d21:=spacecurve({[cos(sud)*(n*cos(u*Pi/2+Pi/2)-2*n)-sin(sud)*(2*n*
sin(u*Pi/2+Pi/2)),sin(sud)*(n*cos(u*Pi/2+Pi/2)-2*n)+cos(sud)*(2*n
*sin(u*Pi/2+Pi/2)),t1],[cos(sud)*(n*cos(u*Pi/2+Pi*3/2)-3*n)-sin
(sud)*(-(2*n*sin(u*Pi/2+Pi*3/2)+2*n)),sin(sud)*(n*cos(u*Pi/2+Pi*3/
2)-3*n)+cos(sud)*(-(2*n*sin(u*Pi/2+Pi*3/2)+2*n)),t1]},u=0..1,style
=line,axes=frame,scaling=constrained,lightmodel=light3,thickness=3,c
olor=red):
d22:=rotate(d21,0,0,pi/2):
d23:=rotate(d21,0,0,-pi/2):
d24:=rotate(d21,0,0,pi):
d6:=display([d21,d22,d23,d24]):

```

```

display([a,d3]):
display([a,d5]):
display([a,d5,b,d6]):
>d11:=plot3d({[v*(n*cos(u*Pi/2+Pi/2)-2*n)+(1-v)*(cos(sud)*(n*cos(
u*Pi/2+Pi/2)-2*n)-sin(sud)*(2*n*sin(u*Pi/2+Pi/2))),v*(2*n*sin(u*Pi
/2+Pi/2)+(1-v)*(sin(sud)*(n*cos(u*Pi/2+Pi/2)-2*n)+cos(sud)*(2*n*
sin(u*Pi/2+Pi/2))),v*t0+(1-v)*t1],[v*(n*cos(u*Pi/2+Pi*3/2)-3*n)+
(1-v)*(cos(sud)*(n*cos(u*Pi/2+Pi*3/2)-3*n)-sin(sud)*(-(2*n*sin
(u*Pi/2+Pi*3/2)+2*n))),v*(-(2*n*sin(u*Pi/2+Pi*3/2)+2*n))+(1-v)*(
sin(sud)*(n*cos(u*Pi/2+Pi*3/2)-3*n)+cos(sud)*(-(2*n*sin(u*Pi/2+
Pi*3/2)+2*n))),v*t0+(1-v)*t1]},v=0..1,u=0..1,style=patchnograd,
axes=normal,scaling=constrained,lightmodel=light3,thickness=3):
d12:=rotate(d11,0,0,Pi/2):
d13:=rotate(d11,0,0,-Pi/2):
d14:=rotate(d11,0,0,pi):
display([d11,d12,d13,d14],labels=[x,y,z]);

>#puntiran pola kepiting
e1:=spacecurve({[n*cos(u*Pi/2+Pi/2)-2*n,(n*sin(u*Pi/2+Pi/2)+n),t0]
,[n/2*cos(u*Pi/2)-3*n,(n*sin(u*Pi/2)+0*n),t0],[n*cos(u*Pi/2+Pi/2)-
2*n,-(n*sin(u*Pi/2+Pi/2)+n),t0],[n/2*cos(u*Pi/2)-3*n,-(n*sin(u*Pi
/2)+0*n),t0]},u=0..1,style=line,axes=frame,scaling=constrained,light
model=light3,thickness=3,color=red):
e2:=rotate(e1,0,0,Pi/2):
e3:=rotate(e1,0,0,-Pi/2):
e4:=rotate(e1,0,0,pi):
e5:=display([e1,e2,e3,e4]):
e21:=spacecurve({[cos(sud)*(n*cos(u*Pi/2+Pi/2)-2*n)-sin(sud)*(n*
sin(u*Pi/2+Pi/2)+n),sin(sud)*(n*cos(u*Pi/2+Pi/2)-2*n)+cos(sud)*(n
*sin(u*Pi/2+Pi/2)+n),t1],[cos(sud)*(n/2*cos(u*Pi/2)-3*n)-sin(sud)*
(n*sin(u*Pi/2)+0*n),sin(sud)*(n/2*cos(u*Pi/2)-3*n)+cos(sud)*(n*sin
(u*Pi/2)+0*n),t1],[cos(sud)*(n*cos(u*Pi/2+Pi/2)-2*n)-sin(sud)*(-(n
*sin(u*Pi/2+Pi/2)+n)),sin(sud)*(n*cos(u*Pi/2+Pi/2)2*n)+cos(sud)*(-
(n*sin(u*Pi/2+Pi/2)+n)),t1],[cos(sud)*(n/2*cos(u*Pi/2)-3*n)-sin
(sud)*(-(n*sin(u*Pi/2)+0*n)),sin(sud)*(n/2*cos(u*Pi/2)3*n)+cos
(sud)*(-(n*sin(u*Pi/2)+0*n)),t1]},u=0..1,style=line,axes=frame,sca
ling=constrained,lightmodel=light3,thickness=3,color=red):
e22:=rotate(e21,0,0,Pi/2):
e23:=rotate(e21,0,0,-Pi/2):
e24:=rotate(e21,0,0,pi):
e6:=display([e21,e22,e23,e24]):
display([a,e3]):
display([a,e5]):
display([a,e5,b,e6]):
>e11:=plot3d({[v*(n*cos(u*Pi/2+Pi/2)-2*n)+(1-v)*(cos(sud)*(n*cos(
u*Pi/2+Pi/2)-2*n)-sin(sud)*(n*sin(u*Pi/2+Pi/2)+n)),v*(n*sin(u*
Pi/2+Pi/2)+n)+(1-v)*(sin(sud)*(n*cos(u*Pi/2+Pi/2)-2*n)+cos(sud)*
(n*sin(u*Pi/2+Pi/2)+n)),v*t0+(1-v)*t1],[v*(n/2*cos(u*Pi/2)-3*n)
+(1-v)*(cos(sud)*(n/2*cos(u*Pi/2)-3*n)-sin(sud)*(n*sin(u*Pi/2)
+0*n)),v*(n*sin(u*Pi/2)+0*n)+(1-v)*(sin(sud)*(n/2*cos(u*Pi/2)-
3*n)+cos(sud)*(n*sin(u*Pi/2)+0*n)),v*t0+(1-v)*t1],[v*(n*cos(u*Pi/2
+Pi/2)-2*n)+(1-v)*(cos(sud)*(n*cos(u*Pi/2+Pi/2)-2*n)-sin(sud)*(-
(n*sin(u*Pi/2+Pi/2)+n))),v*(-(n*sin(u*Pi/2+Pi/2)+n))+(1-v)*(sin
(sud)*(n*cos(u*Pi/2+Pi/2)-2*n)+cos(sud)*(-(n*sin(u*Pi/2+Pi/2)+n))
),v*t0+(1-v)*t1],[v*(n/2*cos(u*Pi/2)-3*n)+(1-v)*(cos(sud)*(n/2*cos
(u*Pi/2)-3*n)-sin(sud)*(-(n*sin(u*Pi/2)+0*n))),v*(-(n*sin(u*Pi/2)
+0*n))+(1-v)*(sin(sud)*(n/2*cos(u*Pi/2)-3*n)+cos(sud)*(-(n*sin(u*

```

```

Pi/2)+0*n))),v*t0+(1-v)*t1}},v=0..1,u=0..1,style=patchnogrid,axes=
normal,scaling=constrained,lightmodel=light3,thickness=3):
e12:=rotate(e11,0,0,Pi/2):
e13:=rotate(e11,0,0,-Pi/2):
e14:=rotate(e11,0,0,Pi):
display([e11,e12,e13,e14]);

```

B. Lampiran Lakop Sapu

B.1 Lakop Pola Lurus

```

> dil:=0.5:
gg:=spacecurve({[P/2,u*(-L/2)+(1-u)*(L/2),t0],[-P/2,u*(-L/2)+(1-
u)*(L/2),t0]},u=0..1,style=line,axes=frame,scaling=constrained,light
model=light3,thickness=3,color=blue):
lkp:=spacecurve({[u*(-P/2)+(1-u)*(P/2),L/2,-P/2],[u*(-P/2)+(1-
u)*(P/2),-L/2,-P/2],[P/2,u*(-L/2)+(1-u)*(L/2),-P/2],[-P/2,u*(-
L/2)+(1-u)*(L/2),-P/2]},u=0..1,style=line,axes=frame,scaling=
constrained,lightmodel=light3,thickness=3,color=red):
lkp1:=translate(lkp,0,0,P/2):
display([lkp,lkp1],labels=[x,y,z]):
>lkp2:=spacecurve({[u*(-P/2*dil)+(1-u)*(P/2*dil),L/2,0],[u*(-
P/2*dil)+(1-u)*(P/2*dil),-L/2,0],[P/2*dil,u*(-L/2)+(1-u)*(L/2),0],[-
P/2*dil,u*(-L/2)+(1-
u)*(L/2),0]},u=0..1,style=line,axes=frame,scaling=constrained,lightm
odel=light3,thickness=3,color=red):
> display([lkp,lkp2],labels=[x,y,z]):
>lakop1:=plot3d({[ (u*(-P/2)+(1-u)*(P/2)),v*(L/2)+(1-v)*(-
L/2),t0],[ (u*(-P/2*dil)+(1-u)*(P/2*dil)),v*(L/2)+(1-v)*(-L/2),t1],
[v*(u*(-P/2)+(1-u)*(P/2))+(1-v)*(u*(-P/2*dil)+(1-
u)*(P/2*dil)),L/2,v*(t0)+(1-v)*(t1)], [v*(u*(-P/2)+(1-u)*(P/2))+(1-
v)*(u*(-P/2*dil)+(1-u)*(P/2*dil)), -L/2,v*(t0)+(1-v)*(t1)],
[v*(u*(-P/2)+(1-u)*(-P/2*dil))+(1-v)*(u*(-P/2)+(1-u)*(-
P/2*dil)),v*(u*(-L/2)+(1-u)*(-L/2))+(1-v)*(u*(L/2)+(1-
u)*(L/2)),u*(t0)+(1-u)*(t1)], [v*(u*(P/2)+(1-u)*(P/2*dil))+(1-
v)*(u*(P/2)+(1-u)*(P/2*dil)),v*(u*(-L/2)+(1-u)*(-L/2))+(1-
v)*(u*(L/2)+(1-u)*(L/2)),u*(t0)+(1-
u)*(t1)]},u=0..1,v=0..1,axes=frame,scaling=constrained,lightmodel=li
ght3,thickness=3):
display(lakop1);

```

B.2 Lakop Pola Lengkung Vertikal

```

>dinding1:=plot3d([u*P/2+(1-u)*(P/2)*dil,v*(L/2)+(1-v)*(-
L/2),(u*t0+(1-
u)*t1)],v=0..1,u=0..1,axes=frame,scaling=constrained,lightmodel=ligh
t3,thickness=3):
dinding2:=plot3d([u*(-P/2)+(1-u)*(-P/2)*dil,v*(L/2)+(1-v)*(-
L/2),(u*t0+(1-
u)*t1)],v=0..1,u=0..1,axes=frame,scaling=constrained,lightmodel=ligh
t3,thickness=3):
dinding:=display([dinding1,dinding2]):
>lakop2_1:=plot3d({[v*((u)^2*(P/2)+2*(1-u)*(u)*(P/4)+(1-
u)^2*(0))+(1-v)*((u)^2*(P/2*dil)+2*(1-u)*(u)*(P/4*dil)+(1-
u)^2*(0)),L/2,(v)*(u)^2*(t0)+2*(1-u)*(u)*(t0+t0/4)+(1-

```

```

u)^2*(t0))+(1-v)*((u)^2*(t1)+2*(1-u)*(u)*(t1+t0/4*dil)+(1-
u)^2*(t1))),
[v*((u)^2*(-P/2)+2*(1-u)*(u)*(-P/4)+(1-u)^2*(0))+(1-v)*((u)^2*(-
P/2*dil)+2*(1-u)*(u)*(-P/4*dil)+(1-
u)^2*(0)),L/2,(v)*((u)^2*(t0)+2*(1-u)*(u)*(t0-t0/4)+(1-
u)^2*(t0))+(1-v)*((u)^2*(t1)+2*(1-u)*(u)*(t1-t0/4*dil)+(1-
u)^2*(t1))]},v=0..1,u=0..1,axes=frame,scaling=constrained,lightmodel
=light3,thickness=3):
lakop2_2:=translate(lakop2_1,0,-L,0):
lakop22:=spacecurve({[(u)^2*(P/2)+2*(1-u)*(u)*(P/4)+(1-
u)^2*(0)),L/2,((u)^2*(t1)+2*(1-u)*(u)*(t1+t0/4)+(1-
u)^2*(t1))],[((u)^2*(-P/2)+2*(1-u)*(u)*(-P/4)+(1-
u)^2*(0)),L/2,((u)^2*(t1)+2*(1-u)*(u)*(t1-t0/4)+(1-
u)^2*(t1))]},u=0..1,axes=frame,scaling=constrained,lightmodel=light3
,thickness=4,color=blue):
lakop222:=translate(lakop22,0,-L,0):
lakop2222:=plot3d({[(u)^2*(P/2)+2*(1-u)*(u)*(P/4)+(1-
u)^2*(0)),v*L/2+(1-v)*(-L/2),((u)^2*(-t)+2*(1-u)*(u)*(-t+t0/4)+(1-
u)^2*(-t))],[((u)^2*(-P/2)+2*(1-u)*(u)*(-P/4)+(1-u)^2*(0)),v*L/2+(1-
v)*(-L/2),((u)^2*(-t)+2*(1-u)*(u)*(-t-t0/4)+(1-u)^2*(-
t))]},v=0..1,u=0..1,axes=frame,scaling=constrained,lightmodel=light3
,thickness=3):
ggg:=spacecurve({[P/2,u*(-L/2)+(1-u)*(L/2),1],[-P/2,u*(-L/2)+(1-
u)*(L/2),1]},u=0..1,style=line,axes=frame,scaling=constrained,lightm
odel=light3,thickness=4,color=blue):
lakop2_3:=plot3d({[(u)^2*(P/2*dil)+2*(1-u)*(u)*(P/4*dil)+(1-
u)^2*(0)),v*L/2+(1-v)*(-L/2),((u)^2*(t1)+2*(1-
u)*(u)*(t1+t0/4*dil)+(1-u)^2*(t1))],[((u)^2*(-P/2*dil)+2*(1-
u)*(u)*(-P/4*dil)+(1-u)^2*(0)),v*L/2+(1-v)*(-L/2),((u)^2*(t1)+2*(1-
u)*(u)*(t1-t0/4*dil)+(1-
u)^2*(t1))]},v=0..1,u=0..1,axes=frame,scaling=constrained,lightmodel
=light3,thickness=3):
lakop2:=display([lakop2_1,lakop2_2,lakop2_3,dinding,lakop2222]):
display([lakop2,w],labels=[x,y,z]):display([lakop22,lakop222,ggg]):

```

B.3 Lakop Pola Lengkung Lurus Vertikal

```

>l21:=spacecurve({[P/5*cos(u*Pi)+P*6/20,L/2,P/10*sin(u*Pi)+t0],
[u*P/10+(1-u)*(-P/10),L/2,t0],
[P/5*cos(u*Pi)-
P*6/20,L/2,P/10*sin(u*Pi)+t0]},u=0..1,style=line,axes=frame,scaling=
constrained,lightmodel=light3,thickness=3,color=blue):
l22:=translate(l21,0,-L,0):
display([l21,l22,gg]):

```

```

>lakop3_1:=plot3d({[v*(P/5*cos(u*Pi)+P*6/20)+(1-
v)*(P/5*dil*cos(u*Pi)+P*6/20*dil),L/2,v*(P/10*sin(u*Pi)+t0)+(1-
v)*(P/10*sin(u*Pi)*dil+t1)], [u*P/10+(1-u)*(-P/10),L/2,v*t0+(1-
v)*(t1)], [v*(P/5*cos(u*Pi)-P*6/20)+(1-v)*(P/5*cos(u*Pi)-
P*6/20)*dil,L/2,v*(P/10*sin(u*Pi)+t0)+(1-
v)*(P/10*sin(u*Pi)*dil+t1)]},v=0..1,u=0..1,axes=frame,scaling=constr
ained,lightmodel=light3,thickness=3):
lakop3_2:=translate(lakop3_1,0,-L,0):
lakop3_3:=plot3d({[(P/5*cos(u*Pi)+P*6/20)*dil,v*L/2+(1-v)*(-
L/2),P/10*dil*sin(u*Pi)+t1],[u*P/10+(1-u)*(-P/10)]*dil,v*L/2+(1-
v)*(-L/2),t1],[(P/5*cos(u*Pi)-P*6/20)*dil,v*L/2+(1-v)*(-

```



```

L/2), P/10*dil*sin(u*Pi)+t1]], v=0..1, u=0..1, style=patchngrid, axes=frame,
scaling=constrained, lightmodel=light3, thickness=3):
lakop333:=plot3d({[(P/5*cos(u*Pi)+P*6/20), v*L/2+(1-v)*(-L/2),
P/10*sin(u*Pi)-t], [(u*P/10+(1-u)*(-P/10)), v*L/2+(1-v)*(-L/2), -
t], [(P/5*cos(u*Pi)-P*6/20), v*L/2+(1-v)*(-L/2), P/10*sin(u*Pi)-
t]], v=0..1, u=0..1, style=patchngrid, axes=frame, scaling=constrained,
lightmodel=light3, thickness=3):
lakop3:=display([lakop3_1, dinding, lakop3_2, lakop3_3, lakop333], labels
=[x, y, z]):
display(lakop3);

```

B.4 Lakop Pola Lengkung Horizontal

```

>
l41:=spacecurve({[P/6*cos(u*Pi)+P/3, P/12*sin(u*Pi)+L/2, t0], [P/6*cos(
u*Pi+Pi)+0, P/22*sin(u*Pi+Pi)+L/2, t0], [P/6*cos(u*Pi)-
P/3, P/12*sin(u*Pi)+L/2, t0]}, u=0..1, style=line, axes=frame, scaling=con
strained, lightmodel=light3, thickness=3, color=blue):
l42:=spacecurve({[P/6*cos(u*Pi+Pi)+P/3, P/12*sin(u*Pi+Pi)-
L/2, t0], [P/6*cos(u*Pi)+0, P/22*sin(u*Pi)-L/2, t0], [P/6*cos(u*Pi+Pi)-
P/3, P/12*sin(u*Pi+Pi)-
L/2, t0]}, u=0..1, style=line, axes=frame, scaling=constrained, lightmodel
=light3, thickness=3, color=blue):
display([l41, l42, gg]):

>
lakop_41:=plot3d({[v*(P/6*cos(u*Pi)+P/3)+(1-
v)*(P/6*cos(u*Pi)+P/3)*dil, v*(P/22*sin(u*Pi)+L/2)+(1-
v)*(P/22*sin(u*Pi)*dil+L/2), v*t0+(1-
v)*(t1)], [v*(P/6*cos(u*Pi+Pi)+0)+(1-
v)*(P/6*cos(u*Pi+Pi)+0)*dil, v*(P/22*sin(u*Pi+Pi)+L/2)+(1-
v)*(P/22*sin(u*Pi+Pi)*dil+L/2), v*t0+(1-v)*(t1)], [v*(P/6*cos(u*Pi)-
P/3)+(1-v)*(P/6*cos(u*Pi)-P/3)*dil, v*(P/22*sin(u*Pi)+L/2)+(1-
v)*(P/22*sin(u*Pi)*dil+L/2), v*t0+(1-
v)*(t1)]], v=0..1, u=0..1, axes=frame, scaling=constrained, lightmodel=li
ght3, thickness=3):

lakop_42:=plot3d({[v*(P/6*cos(u*Pi+Pi)+P/3)+(1-
v)*(P/6*cos(u*Pi+Pi)+P/3)*dil, v*(P/22*sin(u*Pi+Pi)-L/2)+(1-
v)*(P/22*sin(u*Pi+Pi)*dil-L/2), v*t0+(1-
v)*(t1)], [v*(P/6*cos(u*Pi)+0)+(1-
v)*(P/6*cos(u*Pi)+0)*dil, v*(P/22*sin(u*Pi)-L/2)+(1-
v)*(P/22*sin(u*Pi)*dil-L/2), v*t0+(1-v)*(t1)], [v*(P/6*cos(u*Pi+Pi)-
P/3)+(1-v)*(P/6*cos(u*Pi+Pi)-P/3)*dil, v*(P/22*sin(u*Pi+Pi)-L/2)+(1-
v)*(P/22*sin(u*Pi+Pi)*dil-L/2), v*t0+(1-
v)*(t1)]], v=0..1, u=0..1, axes=frame, scaling=constrained, lightmodel=li
ght3, thickness=3):

lakop_43:=plot3d({[(P/6*cos(u*Pi)+P/3)*dil, v*(P/22*sin(u*Pi+Pi)*dil-
L/2)+(1-
v)*(P/22*sin(u*Pi)*dil+L/2), (t1)], [(P/6*cos(u*Pi+Pi)+0)*dil, v*(P/22*
sin(u*Pi)*dil-L/2)+(1-
v)*(P/22*sin(u*Pi+Pi)*dil+L/2), (t1)], [(P/6*cos(u*Pi)-
P/3)*dil, v*(P/22*sin(u*Pi+Pi)*dil-L/2)+(1-
v)*(P/22*sin(u*Pi)*dil+L/2), (t1)]], v=0..1, u=0..1, axes=frame, scaling=
constrained, lightmodel=light3, thickness=3, style=patchngrid):

```

```
lakop444:=plot3d({ [(P/6*cos(u*Pi)+P/3), v*(P/22*sin(u*Pi+Pi)-L/2)+(1-v)*(P/22*sin(u*Pi)+L/2), (-t)], [(P/6*cos(u*Pi+Pi)+0), v*(P/22*sin(u*Pi)-L/2)+(1-v)*(P/22*sin(u*Pi+Pi)+L/2), (-t)], [(P/6*cos(u*Pi)-P/3), v*(P/22*sin(u*Pi+Pi)-L/2)+(1-v)*(P/22*sin(u*Pi)+L/2), (-t)]}, v=0..1, u=0..1, axes=frame, scaling=constrained, lightmodel=light3, thickness=3, style=patchnograd):
```

```
lakop4:=display([lakop_41, lakop_42, lakop_43, lakop444, dinding], labels=[x, y, z]):
display(lakop4);
```

B.5 Lakop Pola Lengkung Lurus Horizontal

```
> l51:=spacecurve({ [P/10*cos(u*Pi)+2*P/5, P/20*sin(u*Pi)+L/2, t0],
[u*(P/2-P/5)+(1-u)*(P/10), L/2, t0],
[P/10*cos(u*Pi), P/20*sin(u*Pi)+L/2, t0],
[u*(-P/2+P/5)+(1-u)*(-P/10), L/2, t0],
[P/10*cos(u*Pi)-2*P/5, P/20*sin(u*Pi)+L/2, t0]}, u=0..1, style=line, axes=frame, scaling=constrained, lightmodel=light3, thickness=3, color=blue):
```

```
l52:=spacecurve({ [P/10*cos(u*Pi+Pi)+2*P/5, P/20*sin(u*Pi+Pi)-L/2, t0],
[u*(P/2-P/5)+(1-u)*(P/10), -L/2, t0],
[P/10*cos(u*Pi+Pi), P/20*sin(u*Pi+Pi)-L/2, t0],
[u*(-P/2+P/5)+(1-u)*(-P/10), -L/2, t0],
[P/10*cos(u*Pi+Pi)-2*P/5, P/20*sin(u*Pi+Pi)-L/2, t0]}, u=0..1, style=line, axes=frame, scaling=constrained, lightmodel=light3, thickness=3, color=blue):
display([l51, l52, gg]):
```

```
> lakop_51:=plot3d({ [v*(P/10*cos(u*Pi)+2*P/5)+(1-v)*(P/10*cos(u*Pi)+2*P/5)*dil, v*(P/20*sin(u*Pi)+L/2)+(1-v)*(P/20*sin(u*Pi)*dil+L/2), v*t0+(1-v)*(t1)],
[v*(u*(P/2-P/5)+(1-u)*(P/10))+(1-v)*(u*(P/2-P/5)+(1-u)*(P/10))*dil, L/2, v*t0+(1-v)*(t1)],
[v*(P/10*cos(u*Pi))+(1-v)*(P/10*cos(u*Pi))*dil, v*(P/20*sin(u*Pi)+L/2)+(1-v)*(P/20*sin(u*Pi)*dil+L/2), v*t0+(1-v)*(t1)],
[v*(u*(-P/2+P/5)+(1-u)*(-P/10))+(1-v)*(u*(-P/2+P/5)+(1-u)*(-P/10))*dil, L/2, v*t0+(1-v)*(t1)],
[v*(P/10*cos(u*Pi)-2*P/5)+(1-v)*(P/10*cos(u*Pi)-2*P/5)*dil, v*(P/20*sin(u*Pi)+L/2)+(1-v)*(P/20*sin(u*Pi)*dil+L/2), v*t0+(1-v)*(t1)]}, u=0..1, v=0..1, axes=frame, scaling=constrained, lightmodel=light3, thickness=3):
```

```
lakop_52:=plot3d({ [v*(P/10*cos(u*Pi+Pi)+2*P/5)+(1-v)*(P/10*cos(u*Pi+Pi)+2*P/5)*dil, v*(P/20*sin(u*Pi+Pi)-L/2)+(1-v)*(P/20*sin(u*Pi+Pi)*dil-L/2), v*t0+(1-v)*(t1)],
[v*(u*(P/2-P/5)+(1-u)*(P/10))+(1-v)*(u*(P/2-P/5)+(1-u)*(P/10))*dil, -L/2, v*t0+(1-v)*(t1)],
[v*(P/10*cos(u*Pi+Pi))+(1-v)*(P/10*cos(u*Pi+Pi))*dil, v*(P/20*sin(u*Pi+Pi)-L/2)+(1-v)*(P/20*sin(u*Pi+Pi)*dil-L/2), v*t0+(1-v)*(t1)],
[v*(u*(-P/2+P/5)+(1-u)*(-P/10))+(1-v)*(u*(-P/2+P/5)+(1-u)*(-P/10))*dil, -L/2, v*t0+(1-v)*(t1)],
```



```
[v*(P/10*cos(u*Pi+Pi)-2*P/5)+(1-v)*(P/10*cos(u*Pi+Pi)-
2*P/5)*dil,v*(P/20*sin(u*Pi+Pi)-L/2)+(1-v)*(P/20*sin(u*Pi+Pi)*dil-
L/2),v*t0+(1-
v)*(t1)]];u=0..1,v=0..1,axes=frame,scaling=constrained,lightmodel=li
ght3,thickness=3):
```

```
lakop_53:=plot3d({[(P/10*cos(u*Pi)+2*P/5)*dil,v*(P/20*sin(u*Pi+Pi)*d
il-L/2)+(1-v)*(P/20*sin(u*Pi)*dil+L/2),(t1)],
[(u*(P/2-P/5)+(1-u)*(P/10))*dil,v*L/2+(1-v)*(-L/2),(t1)],
[(P/10*cos(u*Pi))*dil,v*(P/20*sin(u*Pi+Pi)*dil-L/2)+(1-
v)*(P/20*sin(u*Pi)*dil+L/2),(t1)],
[(u*(-P/2+P/5)+(1-u)*(-P/10))*dil,v*L/2+(1-v)*(-L/2),(t1)],
[(P/10*cos(u*Pi)-2*P/5)*dil,v*(P/20*sin(u*Pi+Pi)*dil-L/2)+(1-
v)*(P/20*sin(u*Pi)*dil+L/2),(t1)]},u=0..1,v=0..1,style=patchnogrid,a
xes=frame,scaling=constrained,lightmodel=light3,thickness=3):
```

```
lakop555:=plot3d({[(P/10*cos(u*Pi)+2*P/5),v*(P/20*sin(u*Pi+Pi)-
L/2)+(1-v)*(P/20*sin(u*Pi)+L/2),(-t)],
[(u*(P/2-P/5)+(1-u)*(P/10)),v*L/2+(1-v)*(-L/2),(-t)],
[(P/10*cos(u*Pi)),v*(P/20*sin(u*Pi+Pi)-L/2)+(1-
v)*(P/20*sin(u*Pi)+L/2),(-t)],
[(u*(-P/2+P/5)+(1-u)*(-P/10)),v*L/2+(1-v)*(-L/2),(-t)],
[(P/10*cos(u*Pi)-2*P/5),v*(P/20*sin(u*Pi+Pi)-L/2)+(1-
v)*(P/20*sin(u*Pi)+L/2),(-
t)]},u=0..1,v=0..1,style=patchnogrid,axes=frame,scaling=constrained,
lightmodel=light3,thickness=6):
```

```
lakop5:=display([lakop_53,lakop_51,lakop_52,dinding,lakop555],labels
=[x,y,z]):
display(lakop5);
```

C. Lampiran Ijuk Sapu

```
> #grid persegi panjang isian lingkaran
```

```
grs1:=plot3d([(v)*((u)*(-3*P/10)+(1-u)*(-3*P/10))+(1-
v)*((u)*(3*P/10)+(1-u)*(3*P/10)),(v)*((u)*(L/2)+(1-u)*(L/2))+(1-
v)*((u)*(L/2)+(1-u)*(L/2)),0],u=0..1,v=0..1,color=blue):
grs2:=translate(grs1,0,-1*n,0):
grs3:=translate(grs1,0,-2*n,0):
grs4:=translate(grs1,0,-3*n,0):
grs5:=translate(grs1,0,-4*n,0):
grs6:=translate(grs1,0,-5*n,0):
grs7:=translate(grs1,0,-6*n,0):
grs11:=plot3d([(v)*((u)*(3*P/10)+(1-u)*(3*P/10))+(1-
v)*((u)*(3*P/10)+(1-u)*(3*P/10)),(v)*((u)*(-L/2)+(1-u)*(-L/2))+(1-
v)*((u)*(L/2)+(1-u)*(L/2)),0],u=0..1,v=0..1,color=blue):
grs12:=translate(grs11,-1*n,0,0):
grs13:=translate(grs11,-2*n,0,0):
grs14:=translate(grs11,-3*n,0,0):
grs15:=translate(grs11,-4*n,0,0):
grs16:=translate(grs11,-5*n,0,0):
grs17:=translate(grs11,-6*n,0,0):
ling1:=plot3d({[1*cos(r)+5,1*sin(r)+2,0*s]},r=0..2*Pi,s=0..1,color=b
lack):
```

```

ling2:=plot3d({[1*cos(r)+5,1*sin(r),0*s]},r=0..2*Pi,s=0..1,color=black):
ling3:=plot3d({[1*cos(r)+5,1*sin(r)-2,0*s]},r=0..2*Pi,s=0..1,color=black):
l:=display([ling1,ling2,ling3]):
ling4:=translate(l,-1*n,0,0):
ling5:=translate(l,-2*n,0,0):
ling6:=translate(l,-3*n,0,0):
ling7:=translate(l,-4*n,0,0):
ling8:=translate(l,-5*n,0,0):
a:=display([grs1,grs2,grs3,grs4,grs11,grs12,grs13,grs14,grs15,grs16,grs17,ling1,ling2,ling3,ling4,ling5,ling6,ling7,ling8]):
display(a);

>dill:=3/2:
grs12:=plot3d([(v)*(u)*(-3*P/10)*dill+(1-u)*(-3*P/10)*dill)+(1-v)*(u)*(3*P/10)*dill+(1-u)*(3*P/10)*dill,(v)*(u)*(L/2)*dill+(1-u)*(L/2)*dill)+(1-v)*(u)*(L/2)*dill+(1-u)*(L/2)*dill,-t],u=0..1,v=0..1,color=blue):
grs22:=translate(grs12,0,-1.5*n,0):
grs32:=translate(grs12,0,-3*n,0):
grs42:=translate(grs12,0,-4.5*n,0):
grs112:=plot3d([(v)*(u)*(3*P/10)*dill+(1-u)*(3*P/10)*dill)+(1-v)*(u)*(3*P/10)*dill+(1-u)*(3*P/10)*dill,(v)*(u)*(-L/2)*dill+(1-u)*(-L/2)*dill)+(1-v)*(u)*(L/2)*dill+(1-u)*(L/2)*dill,-t],u=0..1,v=0..1,color=blue):
grs122:=translate(grs112,-1.5*n,0,0):
grs132:=translate(grs112,-3*n,0,0):
grs142:=translate(grs112,-4.5*n,0,0):
grs152:=translate(grs112,-6*n,0,0):
grs162:=translate(grs112,-7.5*n,0,0):
grs172:=translate(grs112,-9*n,0,0):
ling12:=plot3d({[1*cos(r)*dill+7.5,1*sin(r)*dill+3,0*s-10]},r=0..2*Pi,s=0..1,color=black):
ling22:=plot3d({[1*cos(r)*dill+7.5,1*sin(r)*dill,0*s-10]},r=0..2*Pi,s=0..1,color=black):
ling32:=plot3d({[1*cos(r)*dill+7.5,1*sin(r)*dill-3,0*s-10]},r=0..2*Pi,s=0..1,color=black):
l2:=display([ling12,ling22,ling32]):
a2:=display([grs12,grs22,grs32,grs42,grs112,grs122,grs132,grs142,grs152,grs162,grs172,ling12,ling22,ling32,ling42,ling52,ling62,ling72,ling82]):
ling42:=translate(l2,-1.5*n,0,0):
ling52:=translate(l2,-3*n,0,0):
ling62:=translate(l2,-4.5*n,0,0):
ling72:=translate(l2,-6*n,0,0):
ling82:=translate(l2,-7.5*n,0,0):
b:=translate(a,0,0,t1):
display([a,a2],style=line,axes=frame,scaling=constrained,lightmodel=light3,labels=[x,y,z]):
display([a,b],style=line,axes=frame,scaling=constrained,lightmodel=light3):

```

C.1 Pola Interpolasi Lurus

```
>t2:=t0-1:
```

```

ijuk_1:=spacecurve({[u*(0)+(1-u)*(n),L/2.1,u*t2+(1-u)*(t0-10)],
[u*(n)+(1-u)*(2*n),L/2.1,u*t2+(1-u)*(t0-10)],
[u*(2*n)+(1-u)*(3*n),L/2.1,u*t2+(1-u)*(t0-10)],
[u*(3*n)+(1-u)*(4*n),L/2.1,u*t2+(1-u)*(t0-10)],
[u*(4*n)+(1-u)*(5*n),L/2.1,u*t2+(1-u)*(t0-10)],
[u*(5*n)+(1-u)*(6*n),L/2.1,u*t0+(1-u)*(t0-10)],
[u*(0)+(1-u)*(0),L/2.1,u*t2+(1-u)*(t0-10)],
[u*(0)+(1-u)*(-n),L/2.1,u*t2+(1-u)*(t0-10)],
[u*(-n)+(1-u)*(-2*n),L/2.1,u*t2+(1-u)*(t0-10)],
[u*(-2*n)+(1-u)*(-3*n),L/2.1,u*t2+(1-u)*(t0-10)],
[u*(-3*n)+(1-u)*(-4*n),L/2.1,u*t2+(1-u)*(t0-10)],
[u*(-4*n)+(1-u)*(-5*n),L/2.1,u*t2+(1-u)*(t0-10)],
[u*(-5*n)+(1-u)*(-6*n),L/2.1,u*t0+(1-u)*(t0-
10)]},u=0..1,style=line,axes=none,scaling=constrained,lightmodel=lig
ht3,thickness=3,color=black):
ijuk_2:=translate(ijuk_1,0,-L/2/2,0):
ijuk_3:=translate(ijuk_2,0,-L/2/2,0):
ijuk_4:=translate(ijuk_3,0,-L/2/2,0):
ijuk_5:=translate(ijuk_4,0,-L/2.2/2.2,0):
ijuk1:=display([ijuk_1,ijuk_2,ijuk_3,ijuk_4,ijuk_5]):
> display([lakopl,ijuk1]):

```

C.2 Pola Interpolasi Menyilang

```

> t2:=t0:
ijuk_1:=spacecurve({[u*(0)+(1-u)*(n),L/2,u*t2+(1-u)*(t0-10)],
[u*(n)+(1-u)*(0),L/2,u*t2+(1-u)*(t0-10)],
[u*(2*n)+(1-u)*(3*n),L/2,u*t2+(1-u)*(t0-10)],
[u*(3*n)+(1-u)*(2*n),L/2,u*t2+(1-u)*(t0-10)],
[u*(4*n)+(1-u)*(5*n),L/2,u*t2+(1-u)*(t0-10)],
[u*(5*n)+(1-u)*(4*n),L/2,u*t0+(1-u)*(t0-10)],
[u*(5*n)+(1-u)*(6*n),L/2,u*t0+(1-u)*(t0-10)],

[u*(0)+(1-u)*(n),L/2,u*t2+(1-u)*(t0-10)],
[u*(0)+(1-u)*(-n),L/2,u*t2+(1-u)*(t0-10)],

[u*(0)+(1-u)*(-n),L/2,u*t2+(1-u)*(t0-10)],
[u*(-n)+(1-u)*(0),L/2,u*t2+(1-u)*(t0-10)],
[u*(-2*n)+(1-u)*(-3*n),L/2,u*t2+(1-u)*(t0-10)],
[u*(-3*n)+(1-u)*(-2*n),L/2,u*t2+(1-u)*(t0-10)],
[u*(-4*n)+(1-u)*(-5*n),L/2,u*t2+(1-u)*(t0-10)],
[u*(-5*n)+(1-u)*(-4*n),L/2,u*t0+(1-u)*(t0-10)],
[u*(-5*n)+(1-u)*(-6*n),L/2,u*t0+(1-u)*(t0-10)]
},u=0..1,style=line,axes=none,scaling=constrained,lightmodel=light3,
thickness=3,color=black):
ijuk_2:=translate(ijuk_1,0,-L/2/2,0):
ijuk_3:=translate(ijuk_2,0,-L/2/2,0):
ijuk_4:=translate(ijuk_3,0,-L/2/2,0):
ijuk_5:=translate(ijuk_4,0,-L/2/2,0):

ijuk2:=display([ijuk_1,ijuk_2,ijuk_3,ijuk_4,ijuk_5]):
> display([lakopl,ijuk2],axes=none);

```

C.3 Pola Interpolasi Melingkar/heliks

```
> t2:=t0-1:
ijuk_1:=spacecurve({[cos(u*8*Pi)*(1-u), (1-
u)*sin(u*8*Pi)+L/2,u*t2+(1-u)*(t0-10)],
[cos(u*8*Pi)*(1-u)+n, (1-u)*sin(u*8*Pi)+L/2,u*t2+(1-u)*(t0-10)],
[cos(u*8*Pi)*(1-u)+2*n, (1-u)*sin(u*8*Pi)+L/2,u*t2+(1-u)*(t0-10)],
[cos(u*8*Pi)*(1-u)+3*n, (1-u)*sin(u*8*Pi)+L/2,u*t2+(1-u)*(t0-10)],
[cos(u*8*Pi)*(1-u)+4*n, (1-u)*sin(u*8*Pi)+L/2,u*t2+(1-u)*(t0-10)],
[cos(u*8*Pi)*(1-u)+5*n, (1-u)*sin(u*8*Pi)+L/2,u*t0+(1-u)*(t0-10)],
[cos(u*8*Pi)*(1-u)-n, (1-u)*sin(u*8*Pi)+L/2,u*t2+(1-u)*(t0-10)],
[cos(u*8*Pi)*(1-u)-2*n, (1-u)*sin(u*8*Pi)+L/2,u*t2+(1-u)*(t0-10)],
[cos(u*8*Pi)*(1-u)-3*n, (1-u)*sin(u*8*Pi)+L/2,u*t2+(1-u)*(t0-10)],
[cos(u*8*Pi)*(1-u)-4*n, (1-u)*sin(u*8*Pi)+L/2,u*t2+(1-u)*(t0-10)],
[cos(u*8*Pi)*(1-u)-5*n, (1-u)*sin(u*8*Pi)+L/2,u*t0+(1-u)*(t0-10)]
},u=0..1,style=line,axes=None,scaling=constrained,lightmodel=light3,
thickness=3,color=black):
ijuk_2:=translate(ijuk_1,0,-L/2/2,0):
ijuk_3:=translate(ijuk_2,0,-L/2/2,0):
ijuk_4:=translate(ijuk_3,0,-L/2/2,0):
ijuk_5:=translate(ijuk_4,0,-L/2/2,0):
ijuk3:=display([ijuk_1,ijuk_2,ijuk_3,ijuk_4,ijuk_5]):
> display([lakopl,ijuk3],axes=None):
```