



MODELISASI LIONTIN KALUNG DAN ANTING

SKRIPSI

Oleh

Dzurotul Mutimmah
NIM 071810101087

JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS JEMBER
2012



MODELISASI LIONTIN KALUNG DAN ANTING

SKRIPSI

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

Oleh

Dzurotul Mutimmah
NIM 071810101087

JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS JEMBER
2012

PERSEMBAHAN

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

1. Ibunda Stephany Rosana Palma (Alm.) dan Ayahanda H. Darwono terima kasih atas doa, perhatian, pengorbanan, dan kasih sayang yang telah diberikan.
2. Kakak-kakak tersayang Agnes Narulita Oktoranovia (Alm.) dan Annisa Numratus Tsani, S.E. serta adik saya tersayang Qori'atul Fajriyah yang telah banyak membantu dan memberikan semangat dalam penyelesaian skripsi ini.
3. Guru-guru sejak Taman Kanak-Kanak hingga Perguruan Tinggi, yang telah banyak memberikan ilmu dan membimbing dengan penuh kesabaran.
4. Almamater Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember, SMU Negeri 1 Giri, SLTP Negeri 1 Banyuwangi, SD Negeri Penganjuran VI, dan TK Dharma Wanita.

MOTTO

“Kebenaran itu adalah dari Tuhanmu, sebab itu jangan sekali-kali kamu termasuk orang-orang yang meragu.”

(terjemahan Surat *Al-Baqarah* ayat 7)

”Dan bersabarlah dalam menunggu ketetapan Tuhanmu, maka sesungguhnya kamu berada dalam penglihatan Kami, dan bertasbihlah dengan memuji Tuhanmu ketika kamu bangun berdiri.”

(terjemahan Surat *Ath-Thuur* ayat 48)

PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Dzurotul Mutimmah

NIM : 071810101087

menyatakan dengan sesungguhnya bahwa skripsi yang berjudul "Modelisasi Liontin Kalung dan Anting" adalah benar-benar hasil karya sendiri kecuali jika disebutkan sumbernya dan skripsi ini belum pernah diajukan pada institusi manapun serta bukan karya jiplakan. Saya bertanggung jawab atas keabsahan isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

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

Jember, Januari 2012

Yang menyatakan,

Dzurotul Mutimmah
NIM 071810101087

SKRIPSI

MODELISASI LIONTIN KALUNG DAN ANTING

Oleh

Dzurotul Mutimmah
NIM. 071810101087

Pembimbing

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

Dosen Pembimbing Anggota : Bagus Juliyanto S.Si.

PENGESAHAN

Skripsi berjudul "Modelisasi Liontin Kalung dan Anting" telah diuji dan disahkan pada:

hari :

tanggal :

tempat : Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

Tim Penguji :

Ketua,

Sekretaris,

Prof. Drs. Kusno, DEA., Ph.D.
NIP 196101081986021001

Bagus Juliyanto, S.Si.
NIP 198007022003121001

Anggota I,

Anggota II,

Kosala Dwidja Purnomo, S.Si., M.Si.
NIP 196908281998021001

Kiswara Agung Santoso, M.Kom.
NIP 197209071998031003

Mengesahkan
Dekan,

Prof. Drs. Kusno, DEA., Ph.D.
NIP 196101081986021001

RINGKASAN

Modelisasi Liontin Kalung dan Anting; Dzurotul Mutimmah; 071810101087; 2012; 52 Halaman; Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

Barang perhiasan seperti halnya kalung, anting, gelang, bros, dan pin dapat menambah citra penampilan dan kepercayaan diri seseorang dalam menghadiri acara-acara formal maupun non formal. Bagian utama kalung dan anting yang memberikan keindahan yaitu liontin. Dengan demikian, pemodelan liontin sangat diperlukan untuk mengembangkan variasi model liontin baik dari bentuk, kesimetrian, maupun kesetimbangan ukuran liontin. Penulisan skripsi ini dimaksudkan untuk memodelisasi bentuk liontin yang mencirikan penggabungan benda-benda geometri datar dan geometri ruang.

Dalam penelitian modelisasi liontin ini dibagi menjadi beberapa tahapan. Tahapan pertama adalah memodelisasi liontin pada dimensi dua dengan bentuk dasar segitiga sama kaki. Dalam hal ini membagi daerah segitiga sama kaki menjadi dua bagian yaitu daerah segitiga dan daerah trapesium kemudian mengisi daerah tersebut dengan model-model liontin. Tahapan kedua adalah memodelisasi kerangka liontin cekung dan cembung pada dimensi tiga beralaskan poligon segi enam beraturan dan berketinggian t . Dalam hal ini membagi ketinggian kerangka liontin dan mengisi ketinggian tersebut dengan benda geometri datar dan geometri ruang. Selanjutnya tahapan terakhir dilakukan programasi untuk memodelisasi liontin tersebut dengan bantuan *software* Maple 12.

Hasil penelitian ini didapatkan dua prosedur untuk memodelisasi liontin kalung dan anting, yang pertama prosedur untuk memodelisasi liontin pada dimensi dua dengan bentuk dasar segitiga sama kaki dan kedua untuk memodelisasi kerangka

liontin cekung dan cembung. Prosedur pertama langkah-langkahnya sebagai berikut. Pertama, menetapkan dua buah titik masing-masing terletak pada sisi kaki segitiga sama kaki dan menarik segmen garis melalui kedua buah titik tersebut sehingga membagi daerah segitiga sama kaki menjadi dua bagian, yaitu daerah segitiga dan daerah trapesium. Kedua, membangun pola-pola bentuk liontin, yaitu: (a) membagi ketinggian dan tingkatan pada daerah segitiga kemudian mengisi tiap tingkatan dengan potongan kurva (lingkaran atau elips) sehingga terbangun pola simetri dan bertingkat dan (b) membagi ketinggian dan bagian pada daerah trapesium kemudian mengisi tiap tingkatan dengan potongan kurva (lingkaran atau elips) sehingga terbangun pola simetri dan bertingkat. Sedangkan prosedur kedua langkah-langkahnya sebagai berikut. Pertama, menetapkan jarak titik berat ke titik sudut alas kerangka liontin serta tinggi kerangka liontin tersebut. Kedua, menetapkan jumlah dan jenis benda geometri datar dan geometri ruang pembangun kerangka liontin. Ketiga, membangun potongan-potongan kurva (segmen garis, lingkaran, dan elips) pada bidang XOZ dan YOZ dengan titik ujung-titik ujungnya terletak pada titik sudut benda-benda geometri datar dan geometri ruang.

PRAKATA

Puji syukur kehadirat Allah SWT atas segala rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul "Modelisasi Lontin Kalung dan Anting". 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. Bapak Prof. Drs. Kusno, DEA., Ph.D. selaku Dosen Pembimbing Utama dan Bapak Bagus Juliyanto, S.Si. selaku Dosen Pembimbing Anggota yang telah meluangkan waktu, pikiran, dan perhatian dalam penulisan skripsi ini;
2. Bapak Kosala Dwidja Purnomo, S.Si., M.Si. dan Bapak Kiswara Agung Santoso, M.Kom. selaku Dosen Penguji yang telah memberikan kritikan dan saran demi kesempurnaan skripsi ini;
3. teman-teman angkatan 2007, Dyah, Rona, Risha, Sintia, Rahma, Nurul, Silvi, Landi, Wasil, Hamid, Shandi, Riski, Soraya, Dani, serta teman-teman yang lainnya, terima kasih atas kebersamaan selama waktu kuliah dan telah memberikan semangat serta motivasi;
4. 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, Januari 2012

Penulis

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
HALAMAN PERSEMBAHAN	ii
HALAMAN MOTTO	iii
HALAMAN PERNYATAAN	iv
HALAMAN PEMBIMBINGAN	v
HALAMAN PENGESAHAN	vi
RINGKASAN	vii
PRAKATA	ix
DAFTAR ISI	x
DAFTAR GAMBAR	xii
DAFTAR LAMPIRAN	xv
BAB 1. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan	4
1.4 Manfaat	5
BAB 2. TINJAUAN PUSTAKA	6
2.1 Penyajian Segmen Garis di Bidang dan Segitiga Sama Kaki	6
2.1.1 Penyajian Segmen Garis di Bidang	6
2.1.2 Penyajian Segitiga Sama Kaki	7
2.2 Hitung Sudut di Antara Dua Segmen Garis, Penyajian Poligon Segi Enam Beraturan Beraturan dan Penyajian Belah Ketupat	9
2.2.1 Hitung Sudut di Antara Dua Segmen Garis	9

2.2.2	Penyajian Poligon Segi Enam Beraturan	10
2.2.3	Penyajian Belah Ketupat	12
2.3	Penyajian Lingkaran dan Elips	12
2.4	Penyajian Segmen Garis dan Bidang di Ruang	14
2.4.1	Penyajian Segmen Garis dan Kedudukan Titik pada Segmen Garis di Ruang	14
2.4.2	Penyajian Bidang dan Posisi Titik pada Garis Tegak Lurus Bidang	15
2.4.3	Penyajian Bidang Segi Empat dan Segitiga	18
2.5	Penyajian Limas	19
2.6	Konstruksi Objek pada Program Maple 12	21
BAB 3.	METODE PENELITIAN	27
BAB 4.	HASIL DAN PEMBAHASAN	29
4.1	Modelisasi Lontin pada Dimensi Dua dengan Bentuk Dasar Segitiga Sama Kaki	29
4.1.1	Pola Simetri Sumbu	29
4.1.2	Pola Campuran antara Simetri Sumbu dan Simetri Pusat	42
4.2	Modelisasi Kerangka Lontin Cekung dan Cembung	45
4.3	Pembahasan	47
BAB 5.	KESIMPULAN DAN SARAN	50
5.1	Kesimpulan	50
5.2	Saran	51
DAFTAR PUSTAKA	52
LAMPIRAN	53

DAFTAR GAMBAR

	Halaman
1.1 Beberapa bentuk model liontin kalung dan anting	2
1.2 Tiga titik P_1 , P_2 , dan P_3 pada bidang XOY	3
1.3 Beberapa model liontin dari komposisi elips, lingkaran, dan segitiga	3
1.4 Beberapa model liontin dari komposisi poligon segi enam beraturan, belah ketupat, lingkaran, elips, dan limas	4
2.1 Penyajian segmen garis di bidang	7
2.2 Penyajian segitiga sama kaki	7
2.3 Segitiga siki-siku ABC	8
2.4 Segitiga sama sisi ABC	9
2.5 Ukuran sudut pada dua segmen garis yang saling berpotongan	9
2.6 Poligon segi enam beraturan	10
2.7 Langkah-langkah membangun poligon segi enam beraturan pada bidang $z = z_1$	11
2.8 Poligon segi- n beraturan pada bidang $z = z_1$	11
2.9 Belah ketupat ABC	12
2.10 Penyajian lingkaran dan elips	13
2.11 Penyajian segmen garis diruang	14
2.12 Posisi titik pada segmen garis	15
2.13 Bidang α yang dibentuk dari tiga titik tidak segaris	16
2.14 Posisi titik pada garis tegak lurus bidang	17
2.15 Tahapan pembuatan bidang segi empat	18
2.16 Bidang segitiga dari hasil interpolasi	19
2.17 Limas tegak segi empat $T-ABCD$ dan bagian-bagiannya	19

2.18	Limas tegak	20
2.19	Potongan limas tegak	21
2.20	Segmen garis di ruang pada Maple 12	22
2.21	Lingkaran dengan pusat $O(0,0)$ pada Maple 12.....	22
2.22	Elips dengan pusat $O(0,0)$ pada Maple 12	23
2.23	Bidang pada Maple 12	23
2.24	Bidang segi empat pada Maple 12	24
2.25	Bidang segitiga pada Maple 12	24
2.26	Lingkaran dengan pusat $A(1,1,1)$ pada Maple 12	25
2.27	Keratan lingkaran dengan pusat $A(1,1,1)$ pada Maple 12	25
2.28	Elips dengan pusat $A(1,1,1)$ pada Maple 12	25
2.29	Keratan elips dengan pusat $A(1,1,1)$ pada Maple 12	26
4.1	Segitiga sama kaki pada bidang XOY	30
4.2	Langkah-langkah memodelisasi liontin menggunakan pola trap segitiga sama kaki	32
4.3	Variasi bentuk liontin elips cekung ke atas dan ke bawah untuk pemilihan nilai parameter λ , l , dan t yang berbeda	33
4.4	Variasi bentuk liontin elips cekung ke bawah dan ke atas dengan $\lambda = 1/3$, $\lambda = 1/2$, dan $\lambda = 5/8$	34
4.5	Langkah-langkah memodelisasi liontin menggunakan pola trap segitiga siku-siku	36
4.6	Variasi bentuk liontin menggunakan pola trap segitiga siku-siku untuk $\lambda = 1/3$, $\lambda = 1/2$, dan $\lambda = 5/8$	37
4.7	Langkah-langkah memodelisasi liontin pada trapesium $A_1P_2P_3A_2$	39
4.8	Variasi bentuk liontin elips cekung ke arah kiri untuk $\lambda = 1/3$, $\lambda = 1/2$, dan $\lambda = 5/8$	40
4.9	Variasi bentuk liontin elips cekung ke bawah dan ke arah kanan untuk $\lambda = 1/3$, $\lambda = 1/2$, dan $\lambda = 5/8$	41

4.10 Langkah-langkah memodelisasi liontin pada trapesium $A_1P_2P_3A_2$ dengan pola simetri pusat.....	43
4.11 Variasi bentuk liontin menggunakan pola simetri pusat untuk $\lambda = 1/3$, $\lambda = 1/2$, dan $\lambda = 5/8$	44
4.12 Variasi model liontin dengan bentuk dasar segitiga sama kaki	44
4.13 Contoh langkah-langkah modelisasi liontin dengan bentuk dasar poligon segi enam beraturan	46
4.14 Perubahan cekungan kurva liontin akibat perubahan parameter θ	48
4.15 Validasi bentuk global dari prosedur 4.1 dan 4.2	49

DAFTAR LAMPIRAN

	Halaman
A. Modelisasi Lontin pada Dimensi Dua dengan Bentuk Dasar Segitiga Sama Kaki	53
A.1 Pola Simetri Sumbu	53
A.2 Pola Campuran antara Simetri Sumbu dan Simetri Pusat	57
B. Modelisasi Kerangka Lontin Cekung dan Cembung	58
B.1 Kerangka Lontin Cekung	58
B.2 Kerangka Lontin Cembung	64
B.3 Kerangka Lontin Cekung dan Interpolasi	69

BAB 1. PENDAHULUAN

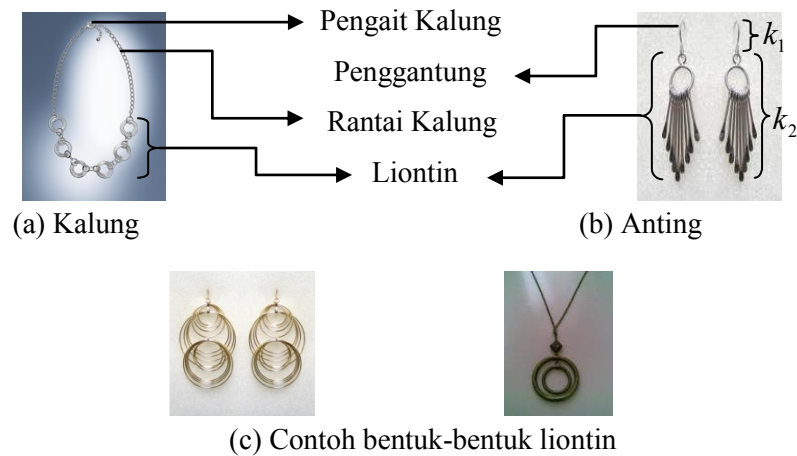
1.1 Latar Belakang

Barang perhiasan seperti halnya kalung, anting, gelang, bros, dan pin dapat menambah citra penampilan dan kepercayaan diri seseorang dalam menghadiri acara-acara formal maupun non formal. Dengan memakai barang perhiasan tersebut, pakaian, kerudung, dan tas yang kita pakai terkesan menjadi lebih indah dan serasi sehingga dapat mempercantik penampilan kita.

Kalung secara umum terdiri atas beberapa bagian yaitu pengait, rantai, dan liontin (Gambar 1.1a). Adapun anting terbangun dari dua komponen yaitu penggantung dan liontin (Gambar 1.1b). Bagian utama kalung dan anting yang memberikan keindahan yaitu liontin. Tampilan liontin yang indah dan unik memberikan nilai jual yang lebih tinggi. Dengan demikian, pemodelan liontin sangat diperlukan untuk mengembangkan variasi model liontin baik dari bentuk, kesimetrian, maupun kesetimbangan ukuran liontin.

Saat ini, model liontin yang telah dibuat berasal dari bermacam-macam bahan, seperti besi, kuningan, monel, emas, dan perak. Bentuknya dibangun dari bermacam-macam benda geometri datar dan ruang, seperti lingkaran dan keratan bola. Dari beberapa bentuk liontin yang telah diperkenalkan terdapat beberapa kelemahan. Pertama, bentuk liontin masih terbangun dari satu gabungan bentuk geometri datar (lingkaran atau elips). Kedua, komponen pembangun liontin umumnya kurang variatif karena masih terdiri dari dua komponen penggantung (k_1 dan k_2) (Gambar 1.1b). Selain itu, jumlah faset mata liontin juga masih terbatas (umumnya permukaan mata liontin mempunyai dua sudut arah refleksi). Sehubungan dengan kendala-kendala bentuk desain liontin tersebut, skripsi ini dimaksudkan untuk memodelisasi

bentuk liontin yang mencirikan penggabungan benda-benda geometri datar dan geometri ruang khususnya komposisi segitiga, poligon segi enam beraturan, belah ketupat, lingkaran, elips, dan keratan limas. Alasan pemilihan bentuk-bentuk geometri datar dan geometri ruang tersebut yaitu dari bentuk geometri yang paling sederhana dapat dimodelisasi menjadi bentuk liontin yang simetris. Selain itu bentuk liontin lebih inovatif karena terdapat benda polihedron, yaitu limas.

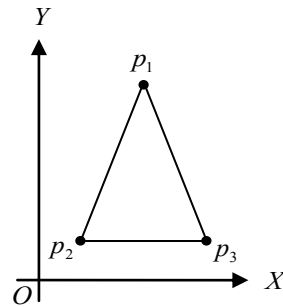


Gambar 1.1 Beberapa bentuk model liontin kalung dan anting

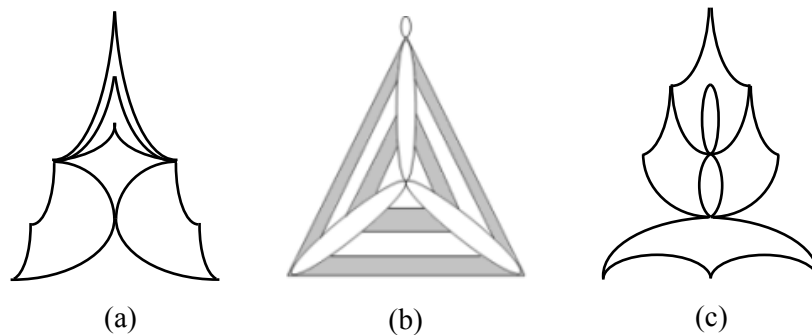
1.2 Perumusan Masalah

Dari beberapa kendala yang dijelaskan pada bagian latar belakang diajukan permasalahan modelisasi liontin sebagai berikut.

- Diberikan tiga titik P_1 , P_2 , dan P_3 pada bidang XOY yang membangun segitiga sama kaki dengan titik puncak di P_1 (Gambar 1.2). Dari segitiga sama kaki tersebut bagaimana prosedur membangun model-model bentuk liontin dari permukaan dimensi dua yang bercirikan kesimetrian dan kesebangunan yang terkomposisi dari bangun elips, lingkaran, dan segitiga sehingga liontin lebih bervariasi dan simetris (Gambar 1.3).

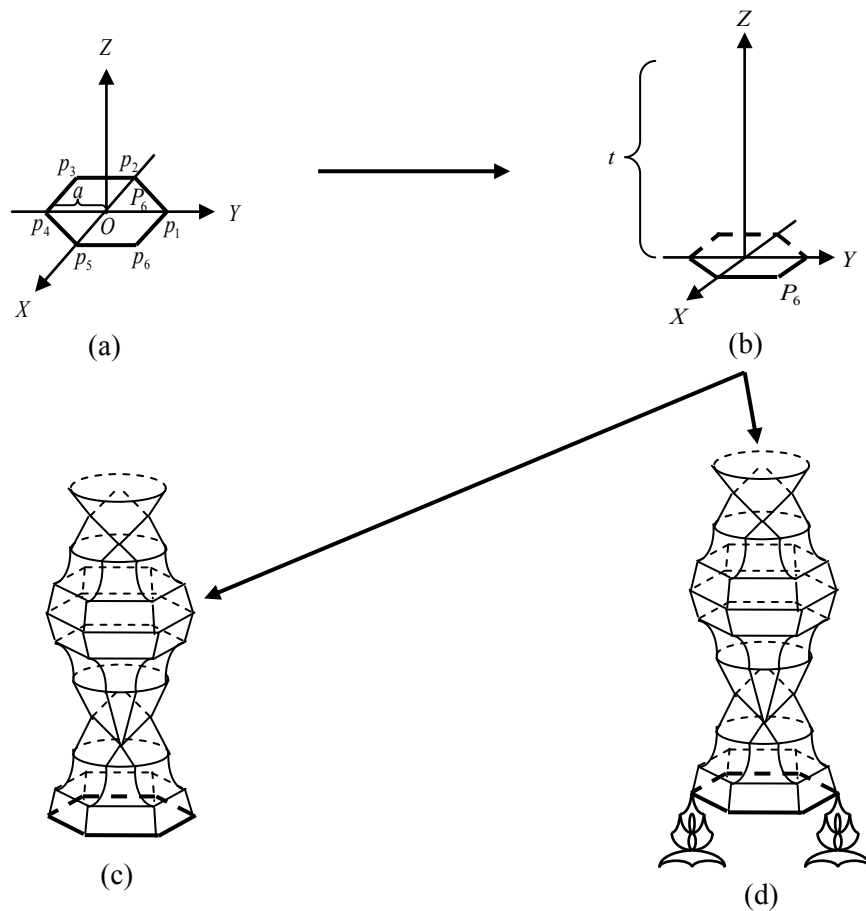


Gambar 1.2 Tiga titik P_1 , P_2 , dan P_3 pada bidang XOY



Gambar 1.3 Beberapa model liontin dari komposisi elips, lingkaran, dan segitiga

- b. Diberikan segi enam beraturan \mathbb{P}_6 dengan pusat di $O(0,0,0)$ dengan titik sudut-titik sudutnya P_1, P_2, P_3, P_4, P_5 , dan P_6 pada bidang XOY , sedangkan a adalah jarak titik berat O ke titik sudut poligon, dipilih $1 \leq a \leq 2$ cm. Dari bentuk segi enam beraturan tersebut bagaimana prosedur membangun beberapa model liontin di dimensi tiga berketinggian t , dimana $3 \leq t \leq 5$ cm, bercirikan kesimetrian yang terkomposisi dari bentuk lingkaran, poligon segi enam beraturan, dan permukaannya multifaset (Gambar 1.4c, d).



Gambar 1.4 Beberapa model liontin dari komposisi poligon segi enam beraturan, belah ketupat, lingkaran, elips, dan limas

1.3 Tujuan

Tujuan dari penelitian ini adalah sebagai berikut:

1. mendapatkan prosedur desain liontin pada dimensi dua dari komposisi elips, lingkaran, dan segitiga dengan bentuk dasar segitiga samakaki;
2. mendapatkan prosedur desain liontin dalam dimensi tiga berbentuk simetris dan variatif terkomposisi dari segitiga, poligon segi enam beraturan, belah ketupat, lingkaran, elips, dan keratan limas simetris vertikal yang beralaskan poligon segi enam beraturan.

1.4 Manfaat

Manfaat dari penelitian ini adalah sebagai berikut:

1. dengan bantuan komputer, dapat dihasilkan beberapa prosedur baru desain liontin yang bervariasi dan simetris;
2. memberikan beberapa daftar model liontin kepada produsen sehingga dapat menambah pilihan model liontin yang sudah ada sebelumnya;
3. dapat meningkatkan nilai jual perhiasan kalung dan anting karena liontin lebih bervariasi.

BAB 2. TINJAUAN PUSTAKA

Sehubungan dengan beberapa permasalahan yang dimaksud dan untuk keperluan mencari solusi permasalahan modelisasi liontin, pada bab ini disajikan beberapa teori dasar yang berkaitan dengan prosedur modelisasi liontin. Teori dasar tersebut meliputi kajian tentang penyajian segmen garis di bidang, segitiga sama kaki, ukuran sudut diantara dua segmen garis, poligon segi enam beraturan, belah ketupat, lingkaran, elips, studi segmen garis dan bidang di ruang, serta benda ruang berupa limas. Hal ini bertujuan untuk mempermudah dalam proses pembuatan modelisasi liontin.

2.1 Penyajian Segmen Garis di Bidang dan Segitiga Sama Kaki

2.1.1 Penyajian Segmen Garis di Bidang

Penyajian segmen garis \overline{AB} yang dibangun oleh dua titik berbeda $A(x_1, y_1)$ dan $B(x_2, y_2)$ di bidang, sebagai titik ujung-titik ujung segmen garis, dapat dinyatakan sebagai tempat kedudukan titik-titik $C(x, y)$ berikut (Gambar 2.1)

$$(1 - \lambda)\langle x_1, y_1 \rangle + \lambda\langle x_2, y_2 \rangle = \langle x, y \rangle, \quad (2.1)$$

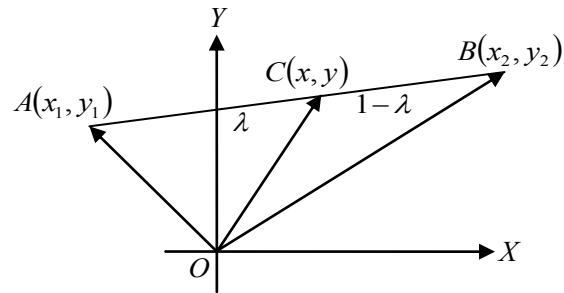
dengan $\lambda \in [0, 1]$.

Bentuk (2.1) juga dapat ditulis menjadi $\begin{bmatrix} x \\ y \end{bmatrix} = (1 - \lambda) \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} + \lambda \begin{bmatrix} x_2 \\ y_2 \end{bmatrix}$, sehingga bentuk persamaan parametriknya adalah

$$\begin{aligned} x(\lambda) &= (1 - \lambda)x_1 + \lambda x_2, \\ y(\lambda) &= (1 - \lambda)y_1 + \lambda y_2. \end{aligned} \quad (2.2)$$

Panjang \overline{AB} dapat diformulasikan sebagai

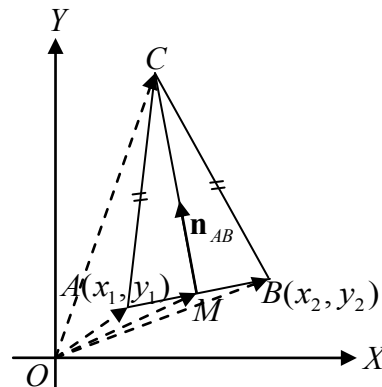
$$d = |\overline{AB}| = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}. \quad (2.3)$$



Gambar 2.1 Penyajian segmen garis di bidang

2.1.2 Penyajian Segitiga Sama Kaki

Misal diberikan dua buah titik sebarang $A(x_1, y_1)$ dan $B(x_2, y_2)$ dibidang XOY (Gambar 2.2). Melalui kedua data titik tersebut dibangun sebuah segitiga sama kaki ABC dengan cara sebagai berikut:



Gambar 2.2 Penyajian segitiga sama kaki

pertama, dibangun segmen garis \overline{AB} melalui persamaan (2.1). Selanjutnya dihitung koordinat titik M yang terletak di tengah-tengah segmen garis \overline{AB} melalui persamaan

$$\overrightarrow{OM} = (\overrightarrow{OA} + \frac{1}{2}\overrightarrow{AB}). \quad (2.4)$$

Kemudian dihitung koordinat titik C melalui persamaan

$$\overrightarrow{OC} = (\overrightarrow{OM} + k\mathbf{n}_{AB}), \quad (2.5)$$

dengan $\mathbf{n}_{AB} = \frac{\langle (y_2 - y_1), -(x_2 - x_1) \rangle}{\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}}$ merupakan normal satuan \overline{AB} dan k adalah bilangan real yang ditetapkan, dimana $k \neq 0$. Selanjutnya membangun segmen garis \overline{AC} dan

\overline{BC} melalui persamaan (2.1) sehingga terbentuk segitiga sama kaki ABC melalui teorema kekongruenan segitiga (S-Sd-S) (Kriswantoro, 2010).

Berdasarkan prosedur membangun segitiga sama kaki ini selanjutnya dapat digunakan untuk membangun segitiga siku-siku dan segitiga sama sisi ABC. Untuk memperoleh segitiga siku-siku ABC (Gambar 2.3), posisi titik C dapat dihitung melalui persamaan:

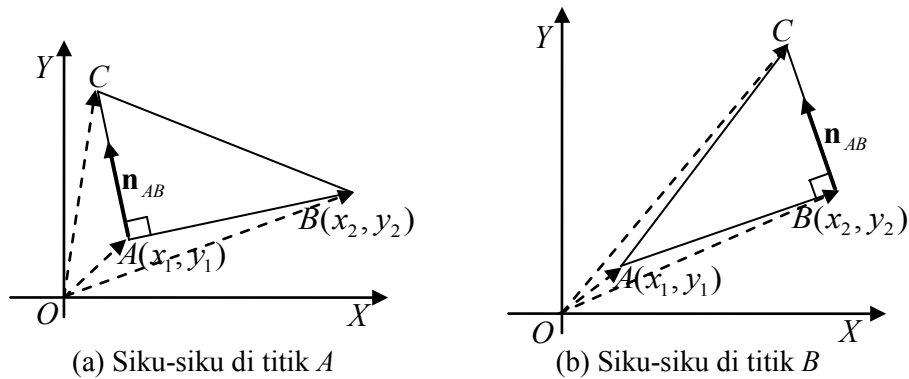
- a. kasus siku-siku di titik A

$$\overline{OC} = (\overline{OA} + k\mathbf{n}_{AB}), \quad (2.6)$$

- b. kasus siku-siku di titik B

$$\overline{OC} = (\overline{OB} + k\mathbf{n}_{AB}). \quad (2.7)$$

Selanjutnya membangun segmen garis \overline{AC} dan \overline{BC} melalui persamaan (2.1).

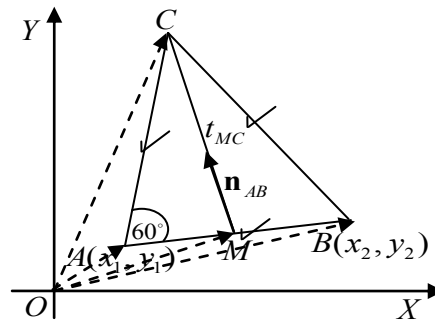


Gambar 2.3 Segitiga siku-siku ABC

Di lain pihak, untuk membangun segitiga sama sisi ABC (Gambar 2.4) dengan $\angle CAM = 60^\circ$ dan tinggi $t_{MC} = |\overline{AM}| \tan 60^\circ = (\sqrt{3})|\overline{AM}|$, posisi titik C dapat dihitung melalui persamaan

$$\overline{OC} = \overline{OM} + t_{MC}\mathbf{n}_{AB}. \quad (2.8)$$

Selanjutnya membangun segmen garis \overline{AC} dan \overline{BC} melalui persamaan (2.1).

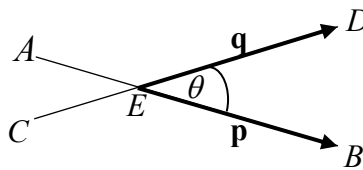


Gambar 2.4 Segitiga sama sisi ABC

2.2 Hitung Sudut di Antara Dua Segmen Garis, Penyajian Poligon Segi Enam Beraturan dan Penyajian Belah Ketupat

2.2.1 Hitung Sudut di Antara Dua Segmen Garis

Setiap dua segmen garis yang saling berpotongan akan membentuk sudut. Titik perpotongan dua segmen garis tersebut dinamakan titik sudut. Sudut diantara dua segmen garis didefinisikan sebagai sudut terkecil yang dibentuk oleh interseksi kedua segmen garis. Misalkan diketahui segmen garis \overline{AB} dan segmen garis \overline{CD} saling berpotongan pada titik E , maka yang dinamakan ukuran sudut kedua segmen garis adalah $\angle BED$ dan $\angle AEC$ dinotasikan θ (Gambar 2.5).



Gambar 2.5 Ukuran sudut pada dua segmen garis yang saling berpotongan

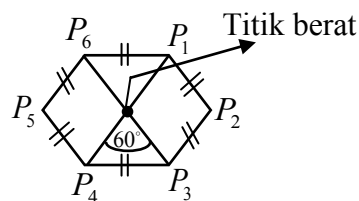
Andaikan pada Gambar 2.5 segmen garis \overline{AB} terdapat vektor $\overline{EB} = \mathbf{p}$ dan pada \overline{CD} terdapat vektor $\overline{ED} = \mathbf{q}$, maka besar sudut θ dapat ditentukan dengan perhitungan:

$$\cos \theta = \frac{\mathbf{p} \cdot \mathbf{q}}{|\mathbf{p}| |\mathbf{q}|},$$

$$\theta = \arccos \frac{\mathbf{p} \cdot \mathbf{q}}{|\mathbf{p}| |\mathbf{q}|} \text{ (Bastian, 2011).}$$

2.2.2 Penyajian Poligon Segi Enam Beraturan

Poligon adalah himpunan titik-titik $P_1, P_2, \dots, P_{n-1}, P_n$ dengan ruas-ruas garis $\overline{P_1P_2}, \overline{P_2P_3}, \dots, \overline{P_{n-1}P_n}, \overline{P_nP_1}$, sedemikian hingga jika dua sebarang ruas garis berpotongan maka akan mempunyai salah satu titik potong dari titik-titik $P_1, P_2, \dots, P_{n-1}, P_n$ dan tidak ada titik lain (Kusno, 2002). Sedangkan poligon segi enam beraturan adalah suatu segi enam dengan panjang sisi dan besar sudut dalam sama, yaitu 120° . Besar sudut pusatnya masing-masing sebesar 60° (Gambar 2.6).



Gambar 2.6 Poligon segi enam beraturan

Berdasarkan definisi poligon segi enam beraturan tersebut, jika diketahui titik beratnya $D(0,0,z_1)$ yang terletak pada bidang $z = z_1$ dan jarak titik $D(0,0,z_1)$ ke titik-titik sudut poligon adalah l , maka dapat dibangun poligon segi enam beraturan dengan langkah-langkah berikut (Gambar 2.7).

- Menetapkan titik sudut poligon awal $P_1(0, l, z_1)$.
- Merotasikan titik $P_1(0, l, z_1)$ terhadap titik berat dengan sudut rotasi sebesar sudut pusat poligon yaitu 60° menggunakan formula

$$\begin{bmatrix} x_{i+1} \\ y_{i+1} \\ z_{i+1} \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} \quad (2.9)$$

dan diperoleh titik $P_2(x_2, y_2, z_1)$.

- Dengan mempertahankan besar sudut 60° dan arah rotasi, mengulangi langkah (b) untuk merotasikan P_2 ke P_3 dan seterusnya hingga dihasilkan titik $P_2(x_2, y_2, z_1), P_3(x_3, y_3, z_1), \dots, P_6(x_6, y_6, z_1)$.
- Membangun poligon segi enam beraturan dengan cara membuat segmen garis diantara dua buah titik sudut yang saling berdekatan, menggunakan formula (Kusno, 2002)

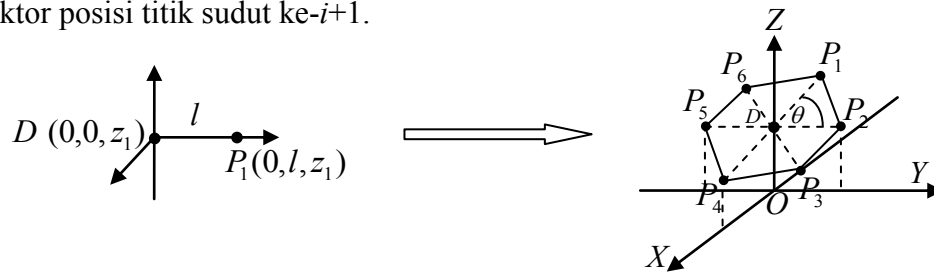
$$(1 - t)\langle x_1, y_1, z_1 \rangle + t\langle x_2, y_2, z_2 \rangle = \langle x, y, z \rangle, \tag{2.10}$$

dengan $t \in [0,1]$. $P_1(x_1, y_1, z_1)$ adalah vektor posisi titik sudut ke-1 dan $P_2(x_2, y_2, z_1)$ vektor posisi titik sudut ke-2. Sedangkan untuk segmen garis pembangun poligon yang lainnya dibangun menggunakan persamaan

$$(1 - t)\langle x_i, y_i, z_i \rangle + t\langle x_{i+1}, y_{i+1}, z_{i+1} \rangle = \langle x, y, z \rangle \text{ untuk } 3 \leq i < 6 \text{ dan}$$

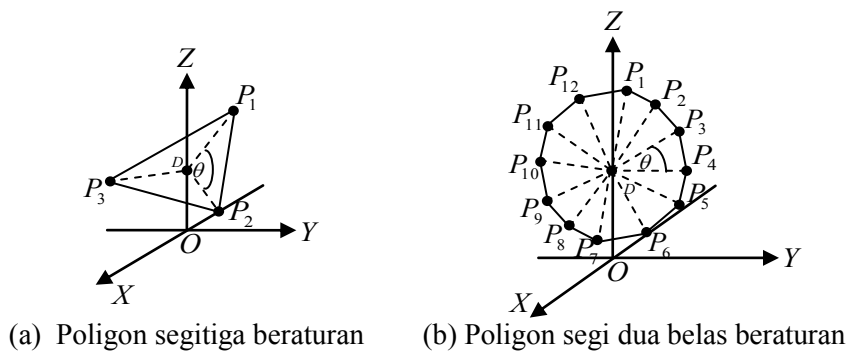
$$(1 - t)\langle x_6, y_6, z_6 \rangle + t\langle x_1, y_1, z_1 \rangle = \langle x, y, z \rangle \text{ untuk } i = 6,$$

dengan $\langle x_i, y_i, z_i \rangle$ adalah vektor posisi titik sudut ke- i dan $\langle x_{i+1}, y_{i+1}, z_{i+1} \rangle$ adalah vektor posisi titik sudut ke- $i+1$.



Gambar 2.7 Langkah-langkah membangun poligon segi enam beraturan pada bidang $z = z_1$

Dari prosedur membangun poligon segi enam ini selanjutnya dapat digunakan untuk membangun poligon segitiga dan segi dua belas beraturan dengan mengubah sudut rotasi pada langkah (b) dan (c) dari ukuran 60° menjadi masing-masing $\frac{360^\circ}{3}$ dan $\frac{360^\circ}{12}$ (Gambar 2.8). Oleh karena itu, untuk poligon segitiga beraturan, besar sudut pusat yang digunakan pada langkah (b) dan (c), yaitu 120° . Sedangkan untuk poligon segi dua belas beraturan, besar sudut pusat yang digunakan pada langkah (b) dan (c), yaitu 30° .



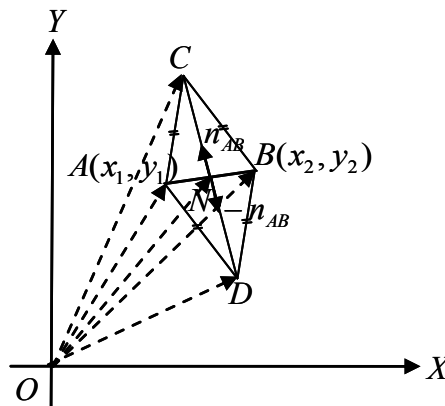
(a) Poligon segitiga beraturan (b) Poligon segi dua belas beraturan

Gambar 2.8 Poligon segi- n beraturan pada bidang $z = z_1$

2.2.3 Penyajian Belah Ketupat

Belah ketupat adalah jajarang genjang dengan dua sisi bersisihannya kongruen (Kusno, 2002). Misal diberikan dua buah titik sebarang $A(x_1, y_1)$ dan $B(x_2, y_2)$ dibidang XOY (Gambar 2.9). Melalui kedua data titik tersebut dibangun sebuah belah ketupat ABCD dengan cara sebagai berikut.

- Membangun segmen garis \overline{AB} melalui persamaan (2.1).
- Menetapkan koordinat titik $N = \left(\frac{x_1+x_2}{2}, \frac{y_1+y_2}{2}\right)$.
- Menghitung koordinat titik C dengan formula $\overrightarrow{OC} = (\overrightarrow{ON} + k\mathbf{n}_{AB})$ dan koordinat titik D dengan formula $\overrightarrow{OD} = (\overrightarrow{ON} - k\mathbf{n}_{AB})$, dengan $\mathbf{n}_{AB} = \frac{\langle (y_2-y_1), -(x_2-x_1) \rangle}{\sqrt{(x_2-x_1)^2 + (y_2-y_1)^2}}$ merupakan normal satuan \overline{AB} dan k adalah bilangan real yang ditetapkan, dengan $k \neq 0$.
- Membangun segmen garis \overline{AC} , \overline{BC} , \overline{AD} , dan \overline{BD} melalui persamaan (2.1) sehingga terbentuk belah ketupat ABCD.



Gambar 2.9 Belah ketupat ABCD

2.3 Penyajian Lingkaran dan Elips

Lingkaran adalah himpunan titik-titik di bidang yang jaraknya terhadap titik tertentu tetap. Titik tetap ini selanjutnya disebut pusat lingkaran (Kusno, 2002). Pada bagian ini akan dijelaskan tentang persamaan parametrik lingkaran. Jika $P(x,y)$

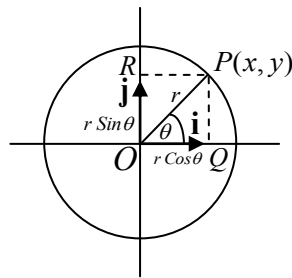
sebarang titik pada lingkaran berpusat di $O(0,0)$, $|\overline{OP}| = r$ dan $\angle POQ = \theta$ maka bentuk persamaan parametrik lingkaran dapat dicari melalui langkah-langkah berikut (Gambar 2.10a).

$$\begin{aligned}\overline{OP} &= \overline{OQ} + \overline{OR}, \\ \langle x - 0, y - 0 \rangle &= \langle r \cos\theta, 0 \rangle + \langle 0, r \sin\theta \rangle, \\ \langle x, y \rangle &= \langle r \cos\theta, r \sin\theta \rangle \text{ sehingga} \\ x(\theta) &= r \cos\theta, \\ y(\theta) &= r \sin\theta.\end{aligned}$$

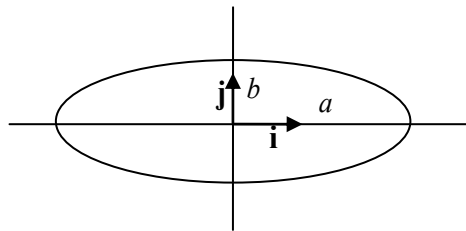
Jadi persamaan parametrik lingkaran berjari-jari r berpusat di $O(0,0)$, yaitu:

$$\mathbf{L}(\theta) = \langle r \cos\theta, r \sin\theta \rangle, \quad (2.11)$$

dengan $0 \leq \theta \leq 2\pi$ dan r adalah suatu konstanta real positif.



(a) Penyajian lingkaran



(b) Penyajian elips

Gambar 2.10 Penyajian lingkaran dan elips

Jika pada persamaan (2.11) nilai parameter r berharga tidak sama untuk arah \mathbf{i} dan arah \mathbf{j} , maka akan diperoleh bentuk elips (Gambar 2.10b). Oleh sebab itu, bentuk persamaan parametrik elips berpusat di $O(0,0)$ yaitu:

$$\mathbf{E}(\theta) = \langle a \cos\theta, b \sin\theta \rangle, \quad (2.12)$$

dengan $0 \leq \theta \leq 2\pi$, sedangkan a dan b adalah suatu konstanta real positif.

2.4 Penyajian Segmen Garis dan Bidang di Ruang

2.4.1 Penyajian Segmen Garis dan Kedudukan Titik pada Segmen Garis di Ruang

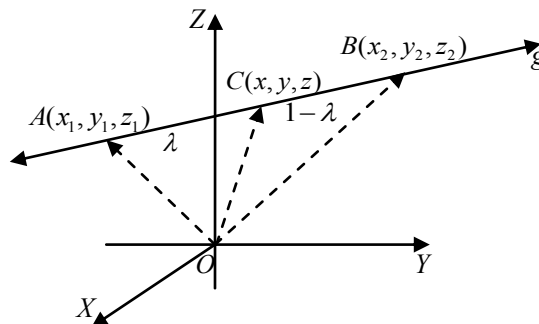
Penyajian segmen garis \overline{AB} yang dibangun oleh dua titik berbeda $A(x_1, y_1, z_1)$ dan $B(x_2, y_2, z_2)$ di ruang sebagai titik ujung-titik ujung segmen garis, dapat dinyatakan sebagai tempat kedudukan titik-titik $C(x, y, z)$ berikut (Gambar 2.11)

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = (1 - \lambda) \begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} + \lambda \begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix}, \quad (2.13)$$

dengan $\lambda \in [0, 1]$ (Budiono, 2011).

Persamaan parametrik dari bentuk (2.13) adalah

$$\begin{aligned} x(\lambda) &= (1 - \lambda)x_1 + \lambda x_2, \\ y(\lambda) &= (1 - \lambda)y_1 + \lambda y_2, \\ z(\lambda) &= (1 - \lambda)z_1 + \lambda z_2. \end{aligned} \quad (2.14)$$



Gambar 2.11 Penyajian segmen garis diruang

Jika diketahui titik C terletak pada segmen garis \overline{PQ} dan membagi \overline{PQ} sehingga $|\overline{PC}| : |\overline{CQ}| = m : n$ (Gambar 2.12a), maka dapat diperoleh koordinat titik C dengan memandang \overline{PQ} sebagai posisi vektor. Jika \mathbf{p} adalah vektor posisi titik P , \mathbf{q} adalah vektor posisi titik Q , dan \mathbf{c} adalah vektor posisi titik C (Gambar 2.12b), maka

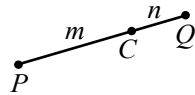
$$\begin{aligned} \overline{PC} : \overline{CQ} = m : n &\rightarrow \overline{PC} = \frac{m}{m+n} \overline{PQ} \text{ dan } \overline{CQ} = \frac{n}{m+n} \overline{PQ}, \\ \mathbf{c} + \overline{CQ} &= \mathbf{q} \rightarrow \mathbf{c} = \mathbf{q} - \overline{CQ}, \\ &\mathbf{c} = \mathbf{q} - \frac{n}{m+n} \overline{PQ}, \\ \overline{PQ} &= \frac{m+n}{n} (\mathbf{q} - \mathbf{c}), \end{aligned} \quad (2.15)$$

$$\mathbf{c} = \mathbf{p} + \overrightarrow{PC} = \mathbf{p} + \frac{m}{m+n} \overrightarrow{PQ}. \quad (2.16)$$

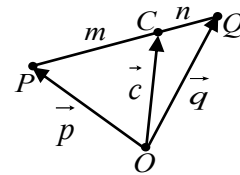
Substitusikan persamaan (2.15) ke persamaan (2.16), didapatkan

$$\begin{aligned} \mathbf{c} &= \mathbf{p} + \frac{m}{m+n} \cdot \frac{m+n}{n} (\mathbf{q} - \mathbf{c}) = \mathbf{p} + \frac{m}{n} (\mathbf{q} - \mathbf{c}), \\ \mathbf{c} \left(\frac{m+n}{n} \right) &= \mathbf{p} + \frac{m}{n} \mathbf{q}, \\ \mathbf{c} &= \frac{n\mathbf{p} + m\mathbf{q}}{m+n}. \end{aligned}$$

Jadi koordinat titik C adalah $\left(\frac{nx_1 + mx_2}{m+n}, \frac{ny_1 + my_2}{m+n}, \frac{nz_1 + mz_2}{m+n} \right)$. (2.17)



(a) Titik C pada \overline{PQ}



(b) Vektor posisi titik P , C , dan Q

Gambar 2.12 Posisi titik pada segmen garis

2.4.2 Penyajian Bidang dan Posisi Titik pada Garis Tegak Lurus Bidang

Bidang dapat dibangun dari tiga buah titik tidak segaris. Misalkan diketahui tiga buah titik $R_1(x_1, y_1, z_1)$, $R_2(x_2, y_2, z_2)$, dan $R_3(x_3, y_3, z_3)$ terletak pada bidang α dan tidak terletak pada satu garis, maka persamaan parametrik bidang α dapat dicari dengan langkah-langkah berikut (Gambar 2.13).

- a. Menghitung dua vektor yang terletak pada bidang α dengan memilih titik $R_1(x_1, y_1, z_1)$ sebagai titik pangkalnya, didapatkan

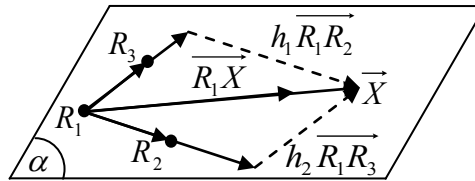
$$\begin{aligned} \overrightarrow{R_1R_2} &= \langle x_2 - x_1, y_2 - y_1, z_2 - z_1 \rangle, \\ \overrightarrow{R_1R_3} &= \langle x_3 - x_1, y_3 - y_1, z_3 - z_1 \rangle. \end{aligned} \quad (2.18)$$

- b. Menghitung vektor $\overrightarrow{R_1X} = \langle x - x_1, y - y_1, z - z_1 \rangle$ ($X = (x, y, z)$) sebagai titik sebarang yang terletak pada bidang α).

- c. Vektor $\overrightarrow{R_1X}$ dapat dinyatakan sebagai bentuk kombinasi linier dari $\overrightarrow{R_1R_2}$ dan $\overrightarrow{R_1R_3}$, yaitu:

$$\begin{aligned}\overrightarrow{R_1X} &= h_1\overrightarrow{R_1R_2} + h_2\overrightarrow{R_1R_3}, \\ \begin{bmatrix} x - x_1 \\ y - y_1 \\ z - z_1 \end{bmatrix} &= h_1 \begin{bmatrix} x_2 - x_1 \\ y_2 - y_1 \\ z_2 - z_1 \end{bmatrix} + h_2 \begin{bmatrix} x_3 - x_1 \\ y_3 - y_1 \\ z_3 - z_1 \end{bmatrix}, \\ x &= x_1 + h_1(x_2 - x_1) + h_2(x_3 - x_1), \\ y &= y_1 + h_1(y_2 - y_1) + h_2(y_3 - y_1), \\ z &= z_1 + h_1(z_2 - z_1) + h_2(z_3 - z_1),\end{aligned}\tag{2.19}$$

dengan h_1 dan h_2 merupakan skalar real. Bentuk persamaan (2.19) disebut sebagai persamaan parametrik bidang α dengan h_1 dan h_2 adalah parameter menggunakan batas $-\infty < h_1, h_2 < \infty$.



Gambar 2.13 Bidang α yang dibentuk dari tiga titik tidak segaris

Vektor normal satuan bidang α (\mathbf{n}_{α_u}) merupakan vektor yang selalu tegak lurus terhadap bidang α dengan panjang satu satuan. Untuk mencari \mathbf{n}_{α_u} dapat dilakukan dengan mengkaliskan dua vektor pada persamaan (2.18) dan membaginya dengan panjang hasil kalisilang dua vektor tersebut, didapatkan

$$\begin{aligned}(\mathbf{n}_{\alpha_u}) &= \frac{\overrightarrow{R_2R_1} \times \overrightarrow{R_2R_3}}{|\overrightarrow{R_2R_1} \times \overrightarrow{R_2R_3}|}, \\ \mathbf{n}_{\alpha_u} &= \left\langle \frac{a}{\sqrt{a^2+b^2+c^2}}, \frac{b}{\sqrt{a^2+b^2+c^2}}, \frac{c}{\sqrt{a^2+b^2+c^2}} \right\rangle,\end{aligned}\tag{2.20}$$

dengan persamaan a , b , dan c sebagai berikut:

$$\begin{aligned}a &= y_1(z_3 - z_2) + y_2(z_1 - z_3) + y_3(z_2 - z_1), \\ b &= x_1(z_2 - z_3) + x_2(z_3 - z_1) + x_3(z_1 - z_2), \\ c &= x_1(y_3 - y_2) + x_2(y_1 - y_3) + x_3(y_2 - y_1).\end{aligned}$$

Di lain pihak jika terdapat titik $R_4(x_4, y_4, z_4)$ juga terletak pada bidang α , maka dapat dicari koordinat titik $S(x_s, y_s, z_s)$ yang terletak pada garis h tegak lurus bidang α dan melalui titik $R_4(x_4, y_4, z_4)$ dengan panjang $\overline{R_4S} = l$ melalui langkah-langkah berikut (Gambar 2.14).

- a. Menghitung vektor satuan $\overline{R_4S}$ ($(\overline{R_4S})_u$), karena $\overline{R_4S}$ dan vektor normal bidang α sejajar maka

$$(\overline{R_4S})_u = \mathbf{n}_{\alpha_u} = \frac{\langle a, b, c \rangle}{\sqrt{a^2 + b^2 + c^2}}.$$

- b. Menghitung vektor posisi titik $S(x_s, y_s, z_s)$, yaitu

$$\overline{R_4S} = l \cdot \mathbf{n}_{\alpha_u},$$

$$\overline{OS} = l \cdot \mathbf{n}_{\alpha_u} + \overline{OR_4},$$

$$\overline{OS} = \left\langle \frac{l \cdot a}{\sqrt{a^2 + b^2 + c^2}} + x_4, \frac{l \cdot b}{\sqrt{a^2 + b^2 + c^2}} + y_4, \frac{l \cdot c}{\sqrt{a^2 + b^2 + c^2}} + z_4 \right\rangle.$$

- c. Menghitung koordinat titik $S(x_s, y_s, z_s)$, karena $\overline{OS}(x_s, y_s, z_s)$ adalah vektor posisi titik $S(x_s, y_s, z_s)$, maka

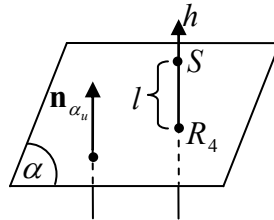
$$S = \left\langle \frac{l \cdot a}{\sqrt{a^2 + b^2 + c^2}} + x_4, \frac{l \cdot b}{\sqrt{a^2 + b^2 + c^2}} + y_4, \frac{l \cdot c}{\sqrt{a^2 + b^2 + c^2}} + z_4 \right\rangle, \quad (2.21)$$

dengan persamaan a , b , dan c sebagai berikut:

$$a = y_1(z_3 - z_2) + y_2(z_1 - z_3) + y_3(z_2 - z_1),$$

$$b = x_1(z_2 - z_3) + x_2(z_3 - z_1) + x_3(z_1 - z_2),$$

$$c = x_1(y_3 - y_2) + x_2(y_1 - y_3) + x_3(y_2 - y_1).$$



Gambar 2.14 Posisi titik pada garis tegak lurus bidang

2.4.3 Penyajian Bidang Segiempat dan Segitiga

Jika diketahui empat buah titik berbeda $S_1(x_1, y_1, z_1)$, $S_2(x_2, y_2, z_2)$, terletak pada garis g_1 dan $S_3(x_3, y_3, z_3)$, $S_4(x_4, y_4, z_4)$ terletak pada garis g_2 dengan $g_1 // g_2$ (Gambar 2.15a), maka dapat dibuat sebuah bidang trapesium (segiempat) dengan titik-titik tersebut sebagai titik sudut bidang menggunakan tahapan berikut (Gambar 2.15).

- Membuat segmen garis dari masing-masing kedua titik tersebut menggunakan persamaan (2.10), sehingga didapatkan dua segmen garis yang sejajar yaitu $\overline{S_1S_2}$ dan $\overline{S_3S_4}$.
- Menginterpolasi kedua segmen garis menggunakan persamaan interpolasi dua kurva (Kusno, 2003)

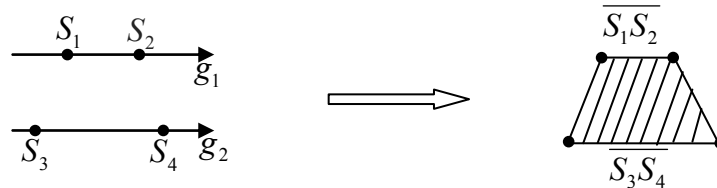
$$S(u, v) = (1 - v)C_1(u) + vC_2(u), \quad (2.22)$$

dengan $C_1(u) = \overline{S_1S_2}(u)$ dan $C_2(u) = \overline{S_3S_4}(u)$, didapatkan

$$S(u, v) = (1 - v)\overline{S_1S_2}(u) + v\overline{S_3S_4}(u), \quad (2.23)$$

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

- Terbangun bidang segiempat $S(u, v)$ dengan titik sudutnya $S_1(x_1, y_1, z_1)$, $S_2(x_2, y_2, z_2)$, $S_3(x_3, y_3, z_3)$, dan $S_4(x_4, y_4, z_4)$.



(a) Posisi titik pada garis yang sejajar

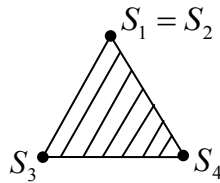
(b) Bidang $S_1S_2S_3S_4$ dari hasil interpolasi

Gambar 2.15 Tahapan pembuatan bidang segi empat

Dalam kasus $S_1 = S_2$, didapatkan sebuah bidang segitiga dalam bentuk (Gambar 2.16)

$$S(u, v) = (1 - v)\overline{OS_1}(u) + v\overline{S_3S_4}(u), \quad (2.24)$$

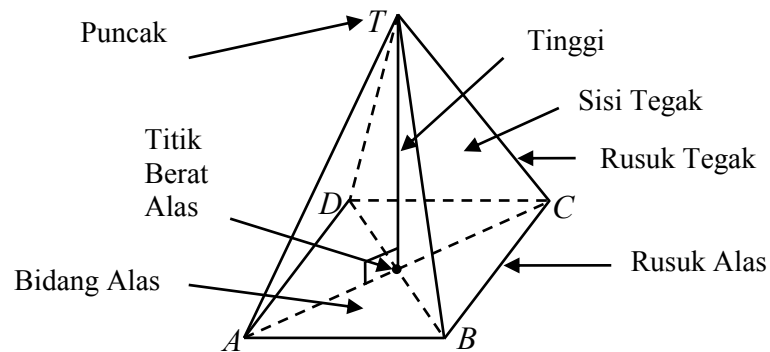
dengan $0 \leq u \leq 1$ dan $0 \leq v \leq 1$, u dan v adalah parameter (Bastian, 2011).



Gambar 2.16 Bidang segitiga dari hasil interpolasi

2.5 Penyajian Limas

Limas adalah suatu bangun ruang yang dibatasi oleh sebuah bidang segi- n (bidang alas) dan n buah segitiga (sisi tegak) yang memiliki satu titik sudut persekutuan (puncak) seperti pada Gambar 2.17. Rusuk-rusuk yang memiliki puncak merupakan rusuk tegak dan sisi dari bidang segi- n merupakan rusuk alas yang membentuk poligon. Suatu limas dikatakan limas tegak jika tingginya adalah dari titik berat alas ke titik puncak limas. Unsur-unsur yang perlu diketahui pada limas dapat dijelaskan pada gambar berikut (Bastian, 2011).

Gambar 2.17 Limas tegak segiempat $T-ABCD$ dan bagian-bagiannya

Jika diketahui persegi panjang (poligon segi empat) atau poligon segi enam beraturan bertitik sudut $P_1(x_1, y_1, z_1)$, $P_2(x_2, y_2, z_2)$, \dots , $P_n(x_n, y_n, z_n)$ dengan $n = 4$ atau $n = 6$, maka dapat dibangun sebuah limas segi- n dengan ketinggian t melalui tahapan berikut (Gambar 2.18).

- a. Menghitung titik perpotongan diagonal alas, yaitu:
 1. untuk limas persegi panjang

titik perpotongan diagonal $P_{d_4} \left(\frac{x_1+x_3}{2}, \frac{y_1+y_3}{2}, \frac{z_1+z_3}{2} \right)$;

2. untuk limas segi enam beraturan

titik perpotongan diagonal $P_{d_6} \left(\frac{x_1+x_4}{2}, \frac{y_1+y_4}{2}, \frac{z_1+z_4}{2} \right)$.

- b. Menghitung posisi titik puncak limas $T(x_t, y_t, z_t)$ dengan ketinggian t dari titik P_d menggunakan persamaan (2.21), didapatkan:

1. untuk limas persegi panjang

$$T(x_t, y_t, z_t) = T \left(\frac{t.a}{\sqrt{a^2+b^2+c^2}} + \frac{x_1+x_3}{2}, \frac{t.b}{\sqrt{a^2+b^2+c^2}} + \frac{y_1+y_3}{2}, \frac{t.c}{\sqrt{a^2+b^2+c^2}} + \frac{z_1+z_3}{2} \right);$$

2. untuk limas segi enam beraturan

$$T(x_t, y_t, z_t) = T \left(\frac{t.a}{\sqrt{a^2+b^2+c^2}} + \frac{x_1+x_4}{2}, \frac{t.b}{\sqrt{a^2+b^2+c^2}} + \frac{y_1+y_4}{2}, \frac{t.c}{\sqrt{a^2+b^2+c^2}} + \frac{z_1+z_4}{2} \right),$$

dengan persamaan a , b , dan c sebagai berikut:

$$a = y_1(z_3 - z_2) + y_2(z_1 - z_3) + y_3(z_2 - z_1),$$

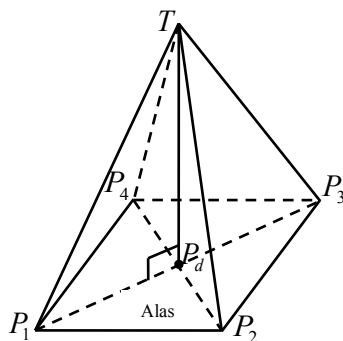
$$b = x_1(z_2 - z_3) + x_2(z_3 - z_1) + x_3(z_1 - z_2),$$

$$c = x_1(y_3 - y_2) + x_2(y_1 - y_3) + x_3(y_2 - y_1).$$

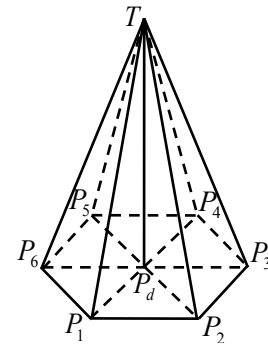
- c. Membangun permukaan limas segi- n , yaitu:

1. menginterpolasi masing-masing rusuk alas terhadap titik $T(x_t, y_t, z_t)$ menggunakan formula (2.24);
2. menginterpolasi segitiga untuk mendapatkan alas limas.

- d. Terbangun limas tegak segi- n seperti pada Gambar 2.18



(a) Persegi panjang



(b) Segi enam beraturan

Gambar 2.18 Limas tegak

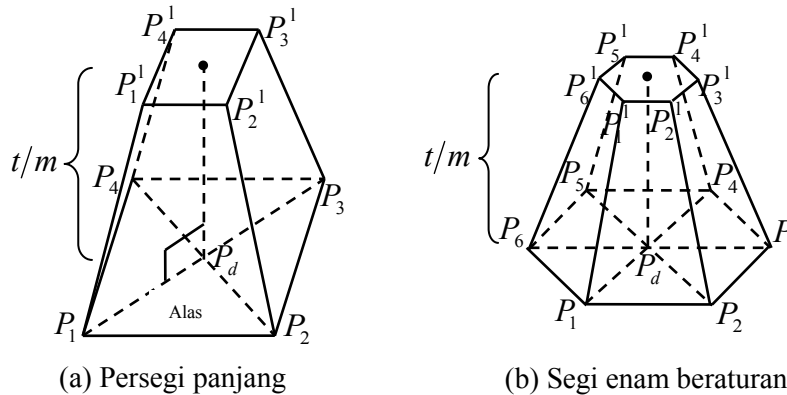
Di lain pihak jika diinginkan suatu potongan limas yang dipotong secara tegak lurus terhadap $\overline{TP_d}$ dengan tinggi t/m dari alas limas, maka dapat dilakukan menggunakan langkah-langkah berikut (Gambar 2.19).

- a. Menghitung koordinat titik sudut bidang potongan limas $P_1^1, P_2^1, \dots, P_n^1$ menggunakan persamaan (2.17), didapatkan

$$P_i^1(x_i^1, y_i^1, z_i^1) = P_i^1 \left(\frac{(m-t).x_i + t.x_t}{m.t}, \frac{(m-t).y_i + t.y_t}{m.t}, \frac{(m-t).z_i + t.z_t}{m.t} \right),$$

dengan $i = 1, 2, \dots, n$. (2.25)

- b. Menginterpolasi segitiga untuk mendapatkan alas atas limas dari hasil potongan dan didapatkan keratan limas tegak seperti pada Gambar 2.19.



Gambar 2.19 Potongan limas tegak

2.6 Konstruksi Objek pada Program Maple 12

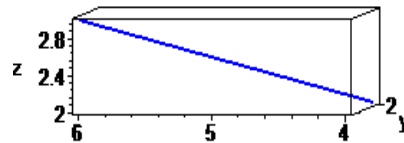
Pada subbab ini disajikan beberapa contoh bahasa pemrograman yang digunakan pada *software* Maple 12 untuk mengkonstruksi objek geometri. Adapun contoh programnya sebagai berikut.

- a. Mengkonstruksi segmen garis

Untuk membangun segmen garis \overline{BC} dengan titik ujung-titik ujungnya adalah $B(4,2,2)$ dan $C(6,3,3)$ pada Maple 12 dengan menggunakan persamaan (2.14), maka dapat dituliskan *script* programnya sebagai berikut:

```
BC:=spacecurve([(1-t)*4+t*6,(1-t)*2+t*3,(1-t)*2+t*3],
t=0..1,axes=frame,labels=[x,y,z],color=blue);
```

Hasil dari program Maple 12 tersebut ditunjukkan pada Gambar (2.20) berikut.



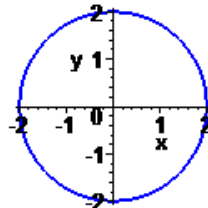
Gambar 2.20 Segmen garis di ruang pada Maple 12

b. Mengkonstruksi lingkaran

Untuk membangun lingkaran yang berpusat di $O(0,0)$ dengan jari-jari 2 satuan pada Maple 12 dengan menggunakan persamaan (2.11), maka dapat dituliskan *script* programnya sebagai berikut:

```
L:=plot([2*cos(t),2*sin(t),t=0..2*Pi],thickness=2,
color=blue,axes=frame,labels=[x,y]);
```

Hasil dari program Maple 12 tersebut ditunjukkan pada Gambar (2.21) berikut.



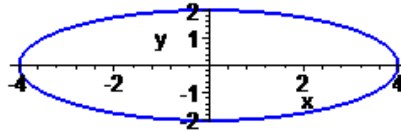
Gambar 2.21 Lingkaran dengan pusat $O(0,0)$ pada Maple 12

c. Mengkonstruksi elips

Untuk membangun elips yang berpusat di $O(0,0)$ dengan panjang mayornya 2×4 satuan dan panjang minornya 2×2 satuan pada Maple 12 dengan menggunakan persamaan (2.12), maka dapat dituliskan *script* programnya sebagai berikut:

```
E:=plot([4*cos(t),2*sin(t),t=0..2*Pi],thickness=2,
color=blue,axes=frame,labels=[x,y]);
```

Hasil dari program Maple 12 tersebut ditunjukkan pada Gambar (2.22) berikut.

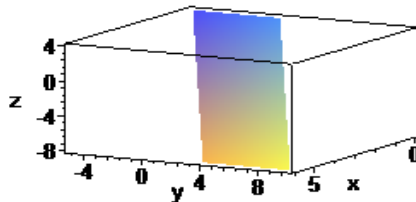
Gambar 2.22 Elips dengan pusat $O(0,0)$ pada Maple 12

d. Mengkonstruksi bidang

Misalkan diberikan titik $A(2,1,0)$, $B(2,4,0)$, dan $C(0,1,4)$ akan dibangun bidang f pada Maple 12 dengan menggunakan persamaan (2.19), maka dapat dituliskan *script* programnya sebagai berikut:

```
f:=plot3d([2+h1*(2-2)+h2*(0-2), 1+h1*(14)+h2*(14),
0+h1*(0-0)+h2*(4-0)], h1=-1..1, h2=2..1, axes=frame,
labels=[x, y, z]);
```

Hasil dari program Maple 12 tersebut ditunjukkan pada Gambar (2.23) berikut.



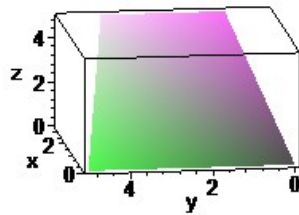
Gambar 2.23 Bidang pada Maple 12

e. Mengkonstruksi bidang segi empat

Misalkan diberikan titik $A(0,0,0)$, $B(0,5,0)$, $C(3,1,5)$ dan $D(3,4,5)$ akan dibangun bidang segi empat pada Maple 12 dengan menggunakan persamaan (2.22), maka dapat dituliskan *script* programnya sebagai berikut:

```
g:=plot3d([(0)*(1-v)+(3)*v, (5*u)*(1-v)+((3*u)+1)*v,
(0)*(1-v)+(5)*v], u=0..1, v=0..1, axes=frame,
labels=[x, y, z]);
```

Hasil dari program Maple 12 tersebut ditunjukkan pada Gambar (2.24) berikut.



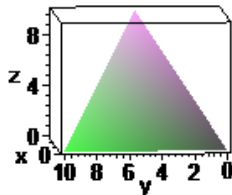
Gambar 2.24 Bidang segi empat pada Maple 12

f. Mengkonstruksi bidang segitiga

Misalkan diberikan titik $A(0,10,0)$, $B(0,0,0)$, dan $C(5,5,10)$ akan dibangun bidang segitiga pada Maple 12 dengan menggunakan persamaan (2.22), maka dapat dituliskan *script* programnya sebagai berikut:

```
h:=plot3d([(0)*(1-v)+(5)*v,(10*u)*(1-v)+(5)*v,(0)*(1-v)+(10)*v],u=0..1,v=0..1,axes=frame,labels=[x,y,z]);
```

Hasil dari program Maple 12 tersebut ditunjukkan pada Gambar (2.25) berikut.



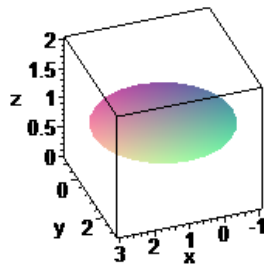
Gambar 2.25 Bidang segitiga pada Maple 12

g. Mengkonstruksi Bidang Lingkaran

Untuk membangun bidang lingkaran yang berpusat di $A(1,1,1)$ dengan jari-jari 2 satuan pada Maple 12, maka dapat dituliskan *script* programnya sebagai berikut:

```
L2:=plot3d([s*2*cos(t)+1,s*2*sin(t)+1,1],s=0..1,t=0..2*Pi,labels=[x,y,z]);
```

Hasil dari program Maple 12 tersebut ditunjukkan pada Gambar (2.26) berikut.

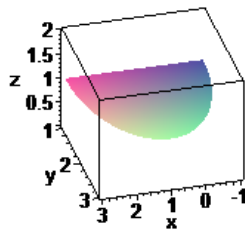


Gambar 2.26 Lingkaran dengan pusat $A(1,1,1)$ pada Maple 12

Sedangkan untuk membangun keratan bidang lingkaran yang berpusat di $A(1,1,1)$, jari-jari 2 satuan, dan bagian keratan pada sudut $0 \leq t \leq \pi$ pada Maple 12, maka dapat dituliskan *script* programnya sebagai berikut:

```
L3:=plot3d([s*2*cos(t)+1, s*2*sin(t)+1, 1], s=0..1, t=0..Pi,
labels=[x, y, z]);
```

Hasil dari program Maple 12 tersebut ditunjukkan pada Gambar (2.27) berikut.



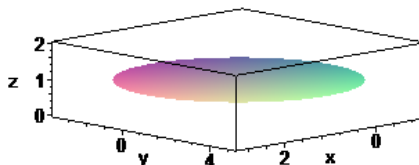
Gambar 2.27 Keratan lingkaran dengan pusat $A(1,1,1)$ pada Maple 12

h. Mengkonstruksi Bidang Elips

Untuk membangun bidang elips yang berpusat di $A(1,1,1)$ dengan panjang mayornya 2×4 satuan dan panjang minornya 2×2 satuan pada Maple 12, maka dapat dituliskan *script* programnya sebagai berikut:

```
E2:=plot3d([s*2*cos(t)+1, s*4*sin(t)+1, 1], s=0..1,
t=0..2*Pi, labels=[x, y, z]);
```

Hasil dari program Maple 12 tersebut ditunjukkan pada Gambar (2.28) berikut.

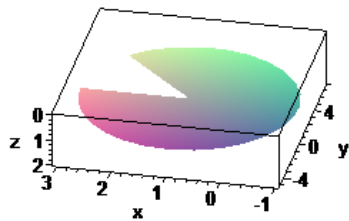


Gambar 2.28 Elips dengan pusat $A(1,1,1)$ pada Maple 12

Sedangkan untuk membangun keratan bidang elips yang berpusat di $A(1,1,1)$ dengan panjang mayornya 2×6 satuan, panjang minornya 2×2 satuan, dan bagian keratan pada sudut $\pi/4 \leq t \leq 2\pi$ pada Maple 12, maka dapat dituliskan *script* programnya sebagai berikut:

```
E4:=plot3d([s*2*cos(t)+1,s*6*sin(t)+1,1],s=0..1,
t=Pi/4..2*Pi,labels=[x,y,z]);
```

Hasil dari program Maple 12 tersebut ditunjukkan pada Gambar (2.29) berikut.



Gambar 2.29 Keratan elips dengan pusat $A(1,1,1)$ pada Maple 12

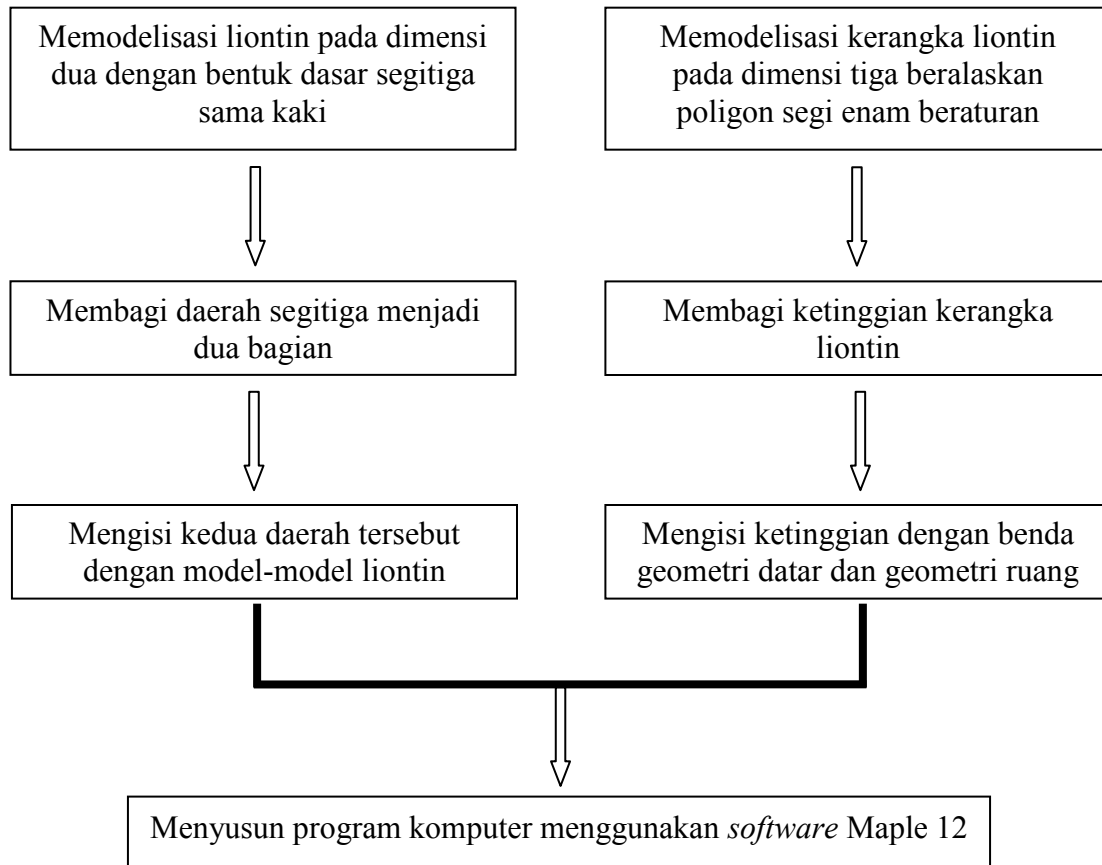
BAB 3. METODE PENELITIAN

Secara umum pelaksanaan penelitian ini meliputi, pertama, menentukan masalah penelitian dan melakukan kajian teori tentang penyajian segmen garis di bidang, segitiga samakaki, ukuran sudut diantara dua segmen garis, poligon segi enam beraturan, belah ketupat, lingkaran, elips, studi segmen garis dan bidang di ruang, serta benda ruang berupa limas. Kedua, mencari prosedur untuk mendapatkan penyelesaian dari masalah yang dirumuskan pada subbab 1.2. Selanjutnya melakukan programasi dengan bantuan *software* Maple 12 dan melakukan simulasi modelisasi liontin kalung dan anting.

Sehubungan dengan langkah kedua tersebut, tahapan kegiatan penelitian adalah sebagai berikut.

1. Memodelisasi liontin pada dimensi dua dengan bentuk dasar segitiga sama kaki. Dalam hal ini membagi daerah segitiga sama kaki menjadi dua bagian yaitu daerah segitiga dan daerah trapesium kemudian mengisi daerah tersebut dengan model-model liontin.
2. Memodelisasi kerangka liontin cekung dan cembung pada dimensi tiga beralaskan poligon segi enam beraturan dan berketinggian t . Dalam hal ini membagi ketinggian kerangka liontin dan mengisi ketinggian tersebut dengan benda geometri datar dan geometri ruang.
3. Menyusun program komputer hasil analisis (1) dan (2) menggunakan *software* Maple 12.

Skema metode penelitian:



BAB 4. HASIL DAN PEMBAHASAN

Pada bab ini dibahas tentang solusi dari masalah modelisasi liontin kalung dan anting. Pertama adalah pembahasan masalah modelisasi liontin pada dimensi dua dengan bentuk dasar segitiga sama kaki tersusun dari potongan kurva (lingkaran atau elips). Kedua adalah pembahasan masalah modelisasi kerangka liontin cekung dan cembung terkomposisi dari belah ketupat, lingkaran, potongan kurva (segmen garis, lingkaran, dan elips) dan keratan limas. Uraian detail di atas dijelaskan sebagai berikut.

4.1 Modelisasi Liontin pada Dimensi Dua dengan Bentuk Dasar Segitiga Sama Kaki

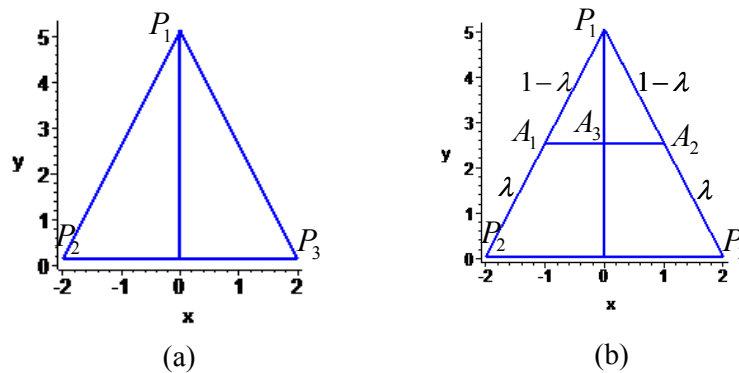
Di bidang XOY diberikan tiga titik $P_1(0, t)$, $P_2(-l, 0)$, dan $P_3(l, 0)$ yang membentuk segitiga sama kaki. Titik $O(0,0)$ sebagai titik tengah segmen garis $\overline{P_2P_3}$ dan tinggi $\overline{OP_1} = t$, dengan $2 \leq t \leq 5$ cm dan $1 \leq |\overline{P_2O}| = |\overline{OP_3}| = l \leq 2$ cm (Gambar 4.1a). Pemilihan nilai t dan l dalam selang tersebut dimaksudkan agar ukuran bentuk liontin proporsional. Berdasarkan data tersebut, didiskusikan langkah-langkah memodelisasi bentuk-bentuk liontin dengan pola simetri sumbu dan pola campuran antara simetri sumbu dan simetri pusat sebagai berikut.

4.1.1 Pola Simetri Sumbu

Prosedur untuk membangun liontin dengan pola simetri sumbu pada segmen garis $\overline{OP_1}$ sebagai berikut:

- menetapkan titik A_1 dan titik A_2 masing-masing pada segmen garis $\overline{P_1P_2}$ dan $\overline{P_1P_3}$ melalui persamaan (2.1), yaitu $\overrightarrow{OA_1} = \lambda\overrightarrow{OP_1} + (1-\lambda)\overrightarrow{OP_2}$ dan $\overrightarrow{OA_2} = \lambda\overrightarrow{OP_1} + (1-\lambda)\overrightarrow{OP_3}$, dengan $0 \leq \lambda \leq 1$ sehingga didapat koordinat titik $A_1((1-\lambda)(-l), \lambda t)$ dan titik $A_2((1-\lambda)(l), \lambda t)$. Dalam kasus ini, dipilih nilai λ yang berbeda, yaitu $1/3 \leq \lambda \leq 5/8$ dan lebih dikhususkan lagi untuk $\lambda = 1/3$, $\lambda = 1/2$, dan $\lambda = 5/8$. Pemilihan λ tersebut dimaksudkan agar ketinggian bentuk liontin proporsional.
- membangun segmen garis $\overline{A_1A_2}$ melalui persamaan (2.1);
- menetapkan posisi titik $A_3(0, \lambda t)$ diperlihatkan pada Gambar 4.1b.

Berdasarkan langkah-langkah tersebut, ada dua bentuk model yang dikembangkan yaitu masing-masing pada segitiga $P_1A_1A_2$ dan trapesium $A_1P_2P_3A_2$.



Gambar 4.1 Segitiga sama kaki pada bidang XOY

a. Kasus Modelisasi Liontin pada Segitiga $P_1A_1A_2$

Untuk memperoleh bentuk-bentuk liontin yang bervariasi, dibangun beberapa pola sebagai berikut.

❖ Pola Trap Segitiga Sama Kaki

- Menetapkan dua buah titik sebarang pada segmen garis $\overline{P_1A_3}$ yaitu titik B_1 dan titik B_2 , sehingga $|\overrightarrow{A_3P_1}| = \overrightarrow{OP_1} - \overrightarrow{OA_3}$, $|\overrightarrow{B_2A_3}| = 1/3 (\overrightarrow{OP_1} - \overrightarrow{OA_3})$, dan $|\overrightarrow{B_1A_3}| = 2/3 (\overrightarrow{OP_1} - \overrightarrow{OA_3})$ (Gambar 4.2a).

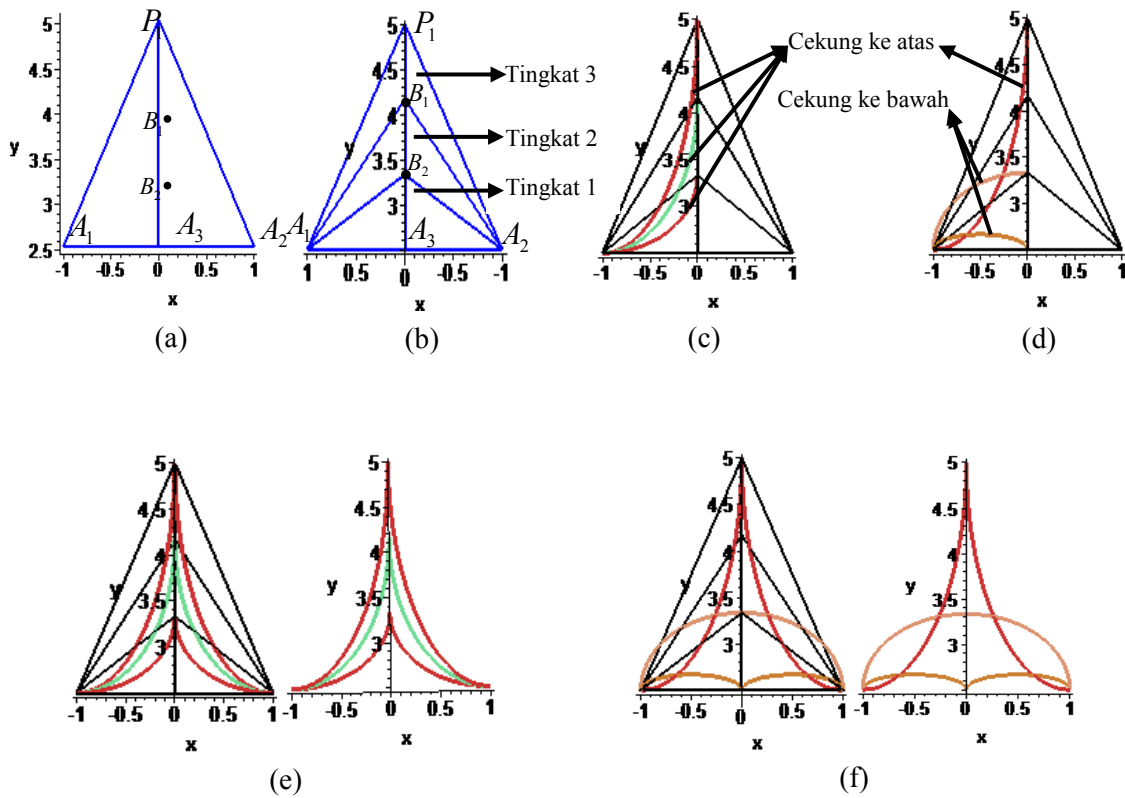
2. Membangun segmen garis $\overline{B_1A_1}$, $\overline{B_2A_1}$, $\overline{B_2A_2}$, dan $\overline{B_1A_2}$ melalui persamaan (2.1) seperti pada Gambar 4.2b.
3. Mengisi tiap tingkatan daerah segitiga dengan bentuk potongan elips atau potongan lingkaran dengan cara antara lain sebagai berikut.

Elips cekung ke atas

- Membangun 1/4 elips cekung ke atas pada tingkatan 1 dengan posisi awal ketinggian pada titik A_1 dan posisi akhir ketinggian pada titik B_2 berpusat di $((1 - \lambda)(-l), (1/3 t_2) + t_2)$, dengan $\lambda = 1/2$, $l = 2$, dan $t_2 = 5/2$.
- Membangun 1/4 elips cekung ke atas pada tingkatan 2 dengan posisi awal ketinggian pada titik A_1 dan posisi akhir ketinggian pada titik B_1 berpusat di $((1 - \lambda)(-l), (2/3 t_2) + t_2)$, dengan $\lambda = 1/2$, $l = 2$, dan $t_2 = 5/2$.
- Membangun 1/4 elips cekung ke atas pada tingkatan 3 dengan posisi awal ketinggian pada titik A_1 dan posisi akhir ketinggian pada titik P_1 berpusat di $((1 - \lambda)(-l), t)$, dengan $\lambda = 1/2$, $l = 2$, dan $t_2 = 5/2$ seperti pada Gambar 4.2c.

Elips cekung ke bawah

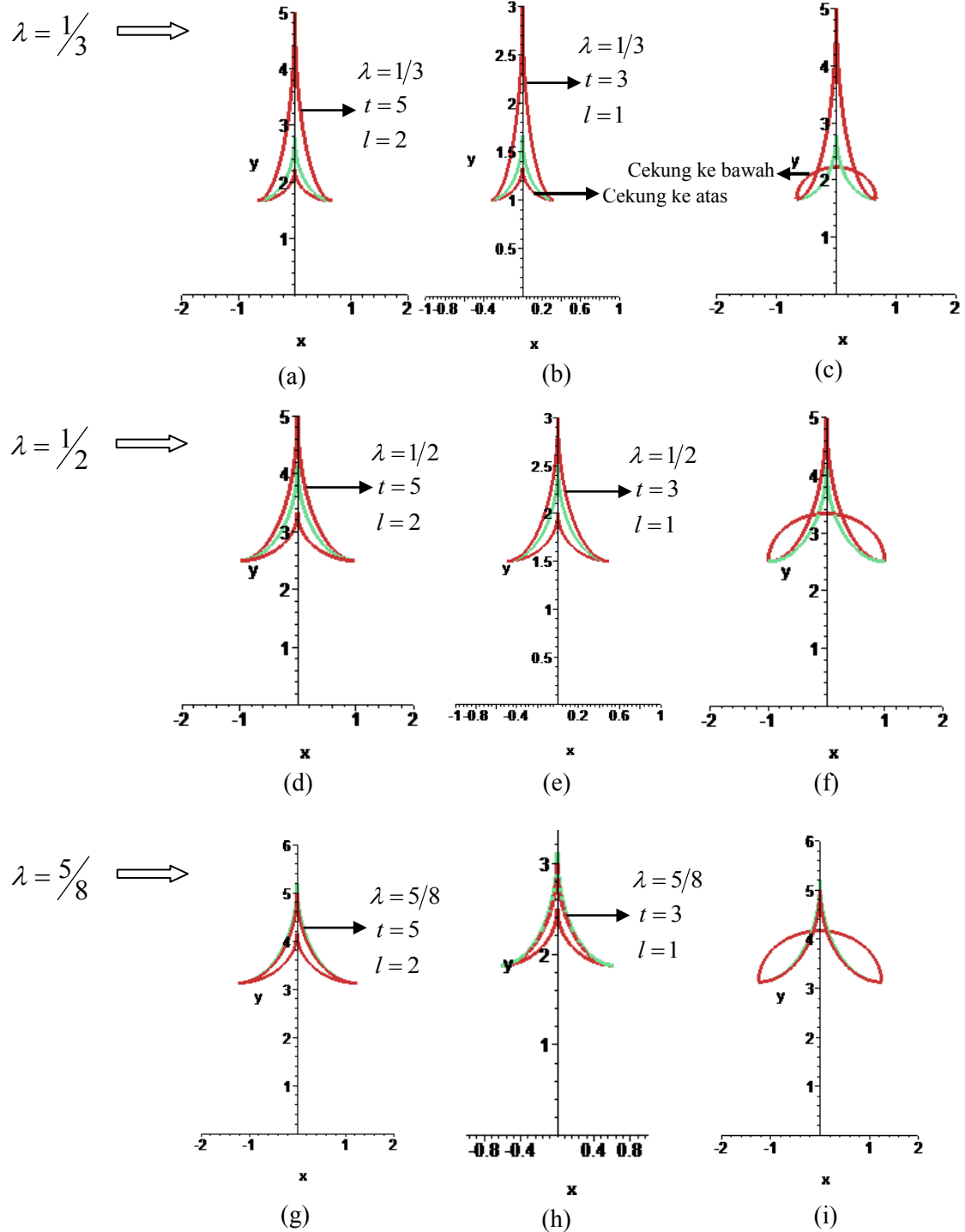
- Membangun 1/2 elips cekung ke bawah pada tingkatan 1 dengan posisi awal ketinggian pada titik A_1 dan posisi akhir ketinggian pada titik A_3 berpusat di $(1/2 (1 - \lambda)(-l), \lambda t)$, dengan $\lambda = 1/2$, $l = 2$, dan $t = 5$.
 - Membangun 1/4 elips cekung ke bawah pada tingkatan 2 dengan posisi awal ketinggian pada titik A_1 dan posisi akhir ketinggian pada titik B_2 berpusat di titik $A_3(0, \lambda t)$, dengan $\lambda = 1/2$ dan $t = 5$.
 - Membangun 1/4 elips cekung ke atas pada tingkatan 3 dengan posisi awal ketinggian pada titik A_1 dan posisi akhir ketinggian pada titik P_1 berpusat di $((1 - \lambda)(-l), t)$, dengan $\lambda = 1/2$, $l = 2$, dan $t = 5$ seperti pada Gambar 4.2d.
4. Refleksikan hasil perlakuan (3) terhadap $\overline{A_3P_1}$ untuk mengisi daerah segitiga $P_1A_2A_3$ (Gambar 4.2e, f).



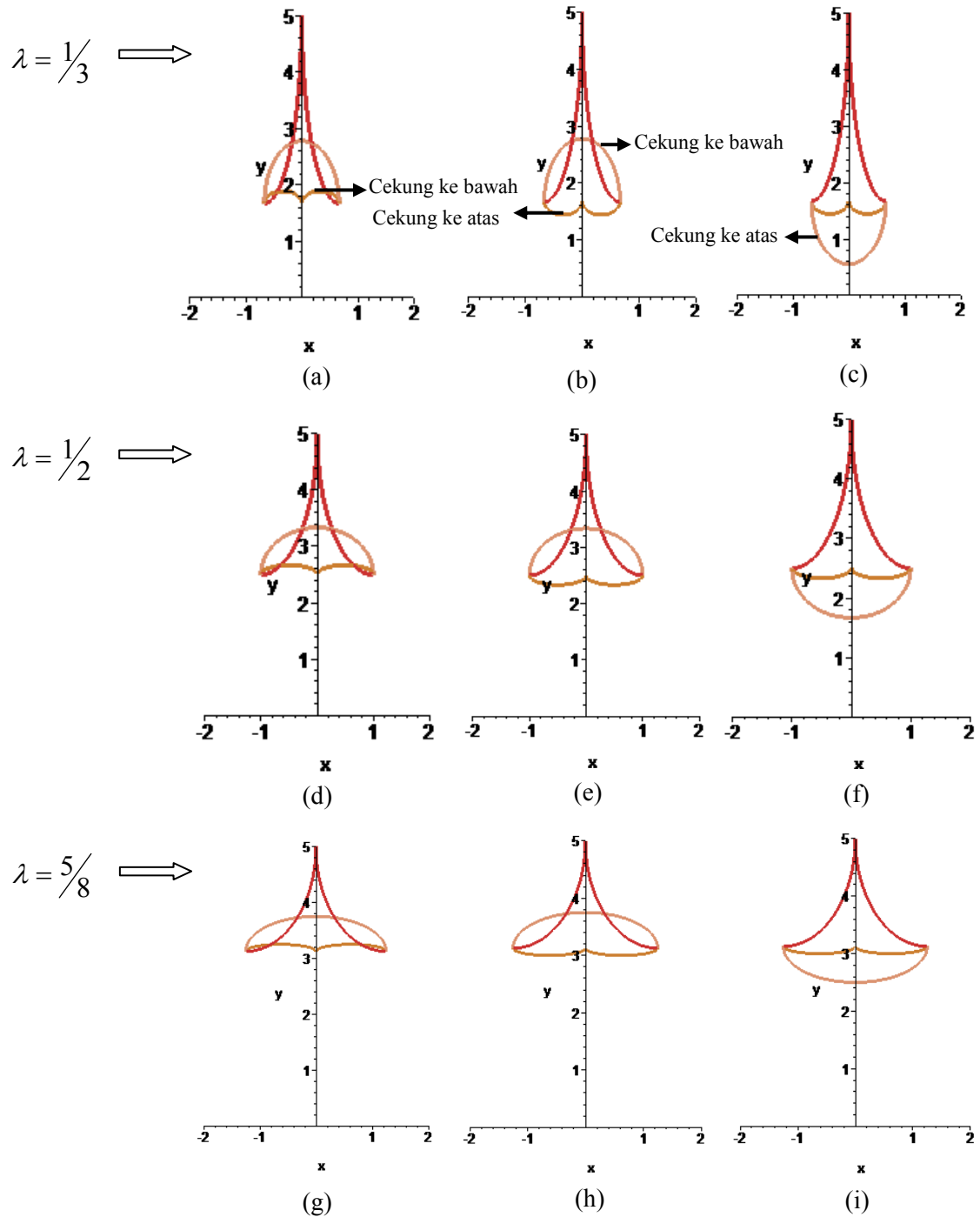
Gambar 4.2 Langkah-langkah memodelisasi liontin menggunakan pola trap segitiga sama kaki

Implementasi desain liontin model cekung ke atas dan cekung ke bawah

Dari prosedur modelisasi liontin yang dibahas dari bagian 4.1.1, selanjutnya dapat dikembangkan beberapa bentuk liontin yang bermacam-macam dengan pengambilan nilai λ yang berbeda, yaitu $\lambda = 1/3$, $\lambda = 1/2$, dan $\lambda = 5/8$ yang hasilnya ditunjukkan pada Gambar 4.3 dan Gambar 4.4 dibawah ini.



Gambar 4.3 Variasi bentuk liontin elips cekung ke atas dan ke bawah untuk pemilihan nilai parameter λ , l , dan t yang berbeda

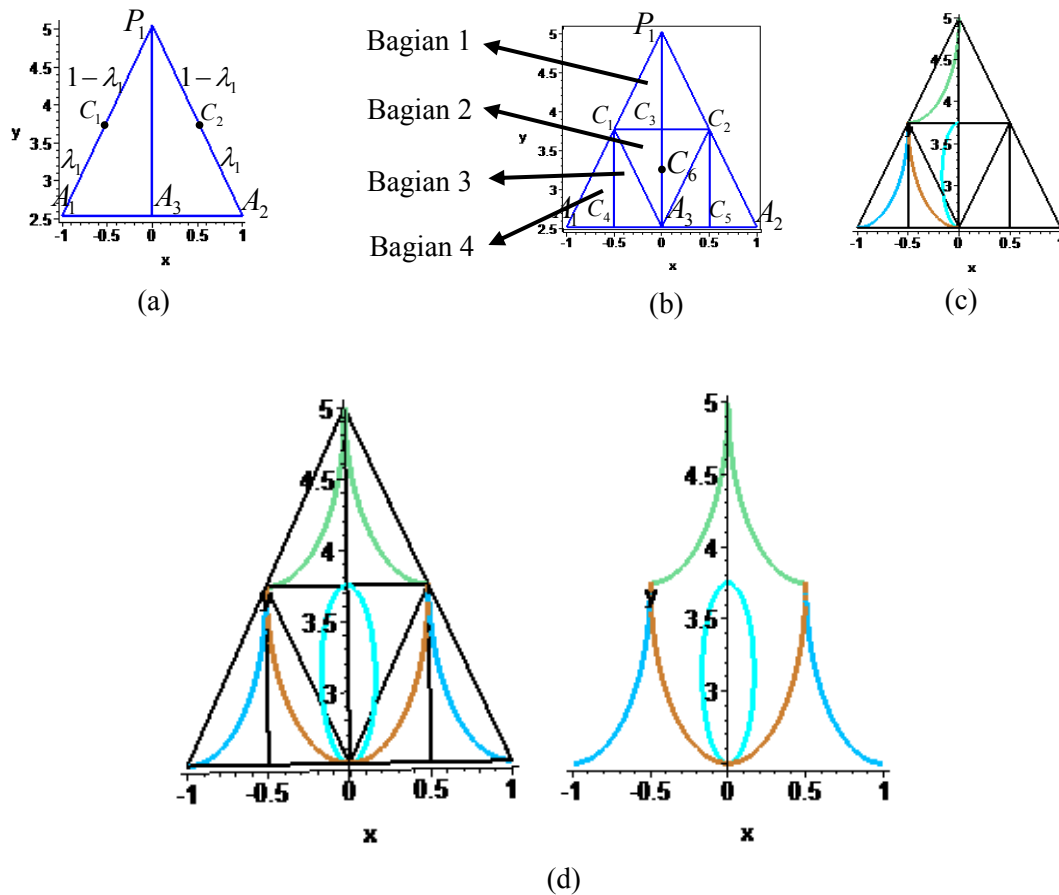


Gambar 4.4 Variasi bentuk liontin elips cekung ke bawah dan ke atas dengan $\lambda = 1/3$, $\lambda = 1/2$, dan $\lambda = 5/8$

❖ Pola Trap Segitiga siku-siku

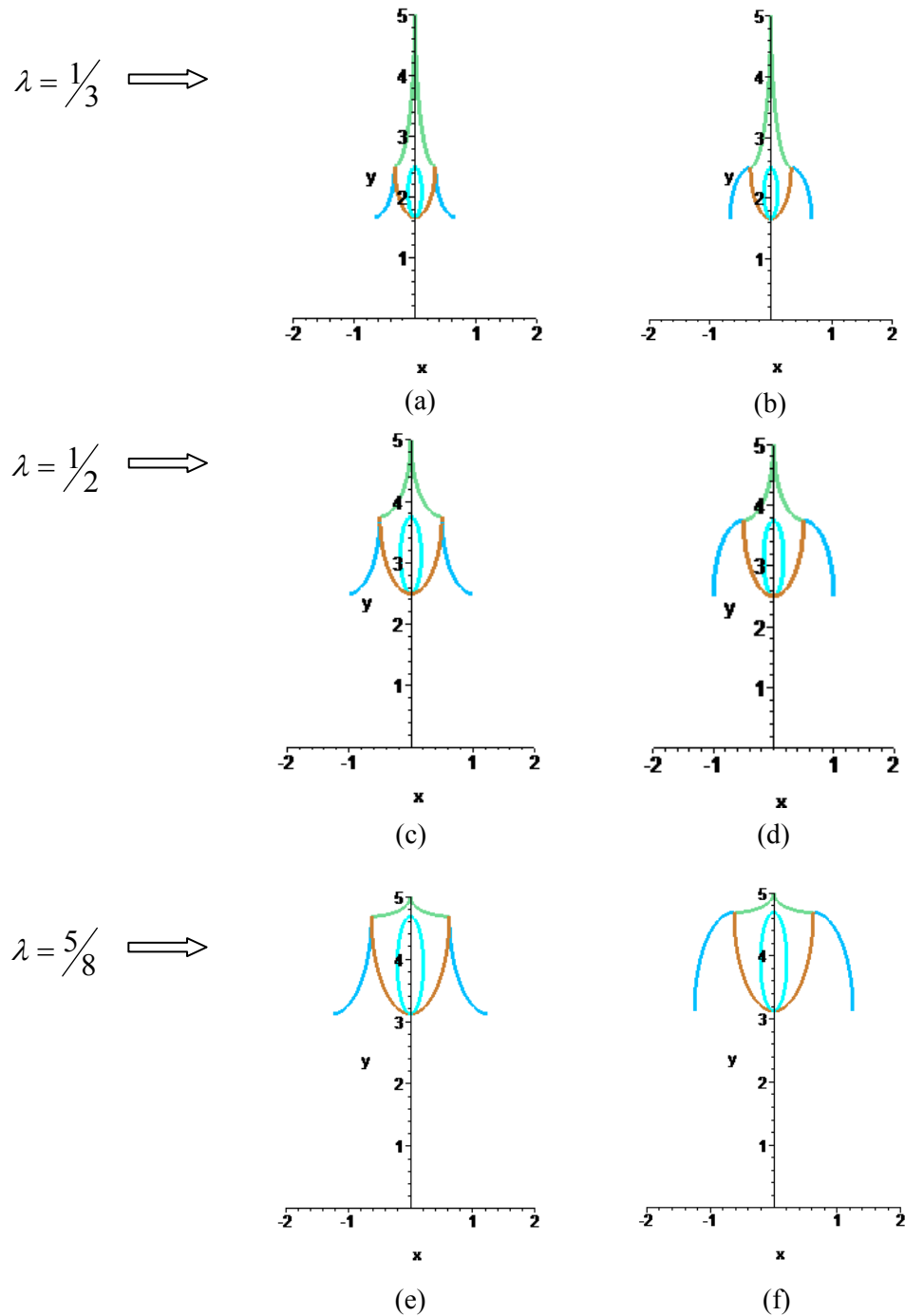
1. Menetapkan posisi titik C_1 dan titik C_2 masing-masing pada segmen garis $\overline{P_1A_1}$ dan $\overline{P_1A_2}$ melalui persamaan (2.1), yaitu $\overline{A_3C_1} = \lambda_1 \overline{A_3P_1} + (1 - \lambda_1) \overline{A_3A_1}$ dan $\overline{A_3C_2} = \lambda_1 \overline{A_3P_1} + (1 - \lambda_1) \overline{A_3A_2}$ sehingga didapat koordinat titik $C_1 \left((1 - \lambda_1)x_{A_1} + \lambda_1 x_{P_1}, (1 - \lambda_1)y_{A_1} + \lambda_1 y_{P_1} \right)$ dan titik $C_2 \left((1 - \lambda_1)x_{A_3} + \lambda_1 x_{P_1}, (1 - \lambda_1)y_{A_3} + \lambda_1 y_{P_1} \right)$, dengan $1/3 \lambda \leq \lambda_1 \leq 5/8 \lambda$ (Gambar 4.5a).
2. Membangun segmen garis $\overline{C_1C_2}$, $\overline{C_1A_3}$, dan $\overline{C_2A_3}$ melalui persamaan (2.1).
3. Menetapkan posisi titik $C_3(x_{A_3}, y_{C_1})$, titik $C_4(x_{C_1}, y_{A_3})$, dan titik $C_5(x_{C_2}, y_{A_3})$.
4. Membangun segmen garis $\overline{C_1C_4}$, dan $\overline{C_2C_5}$ melalui persamaan (2.1) seperti pada Gambar 4.5b.
5. Mengisi tiap bagian daerah cacahan segitiga dengan bentuk potongan elips atau potongan lingkaran dengan tahapan sebagai berikut.
 - Membangun 1/4 elips cekung ke atas pada bagian 1 dengan posisi awal ketinggian pada titik C_1 dan posisi akhir ketinggian pada titik P_1 berpusat di (x_{C_1}, y_{P_1}) .
 - Membangun 1/2 elips cekung ke arah kanan pada bagian 2 dengan posisi awal ketinggian pada titik C_3 dan posisi akhir ketinggian pada titik A_3 berpusat di (x_{A_3}, y_{C_6}) , dengan perhitungan $\overline{OC_6} = \lambda_1 \overline{OP_1} + (1 - \lambda_1) \overline{OA_3}$.
 - Membangun 1/4 elips cekung ke atas pada bagian 3 dengan posisi awal ketinggian pada titik C_1 dan posisi akhir ketinggian pada titik A_3 berpusat di titik $C_3(x_{A_3}, y_{C_1})$.
 - Membangun 1/4 elips cekung ke atas pada bagian 4 dengan posisi awal ketinggian pada titik A_1 dan posisi akhir ketinggian pada titik C_1 berpusat di (x_{A_1}, y_{C_1}) seperti pada Gambar 4.5c.

6. Refleksikan hasil perlakuan (5) terhadap $\overline{A_3P_1}$ untuk mengisi tiap bagian daerah cacahan segitiga $P_1A_2A_3$ (Gambar 4.5d).



Gambar 4.5 Langkah-langkah memodelisasi liontin menggunakan pola trap segitiga siku-siku

Dari Gambar 4.5d, dapat dilihat perbedaan antara model yang satu dengan model lainnya. Selanjutnya, dapat dikembangkan beberapa bentuk liontin yang bermacam-macam dengan pengambilan nilai λ yang berbeda, yaitu $\lambda = 1/3$, $\lambda = 1/2$, dan $\lambda = 5/8$ yang hasilnya ditunjukkan pada Gambar 4.6 dibawah ini.



Gambar 4.6 Variasi bentuk liontin menggunakan pola trap segitiga siku-siku untuk $\lambda = 1/3$, $\lambda = 1/2$, dan $\lambda = 5/8$

b. Kasus Modelisasi Liontin pada Trapesium $A_1P_2P_3A_2$

Untuk memperoleh bentuk-bentuk liontin yang bervariasi, dibangun pola segitiga sepusat sebagai berikut.

1. Menetapkan posisi titik D_1 , dan titik D_2 masing-masing pada segmen garis $\overline{A_1P_2}$ dan $\overline{A_2P_3}$ melalui persamaan (2.1), yaitu $\overrightarrow{OD_1} = \lambda_2 \overrightarrow{OA_1} + (1 - \lambda_2) \overrightarrow{OP_2}$ dan $\overrightarrow{OD_2} = \lambda_2 \overrightarrow{OA_2} + (1 - \lambda_2) \overrightarrow{OP_3}$ sehingga didapat koordinat titik $D_1 \left((1 - \lambda_2)x_{P_2} + \lambda_2 x_{A_1}, (1 - \lambda_2)y_{P_2} + \lambda_2 y_{A_1} \right)$ dan titik $D_2 \left((1 - \lambda_2)x_{P_3} + \lambda_2 x_{A_2}, (1 - \lambda_2)y_{P_3} + \lambda_2 y_{A_2} \right)$ seperti pada Gambar 4.7a, dengan $\lambda_2 = \lambda_1$.
2. Menetapkan posisi titik $D_3(x_{A_3}, y_{D_1})$.
3. Membangun segmen garis $\overline{A_1D_3}$, $\overline{D_1D_3}$, $\overline{P_2D_3}$, $\overline{P_3D_3}$, $\overline{D_2D_3}$, dan $\overline{A_2D_3}$ melalui persamaan (2.1) seperti pada Gambar 4.7b.
4. Mengisi tiap bagian dengan bentuk potongan ellips dan potongan lingkaran dengan tahapan sebagai berikut.

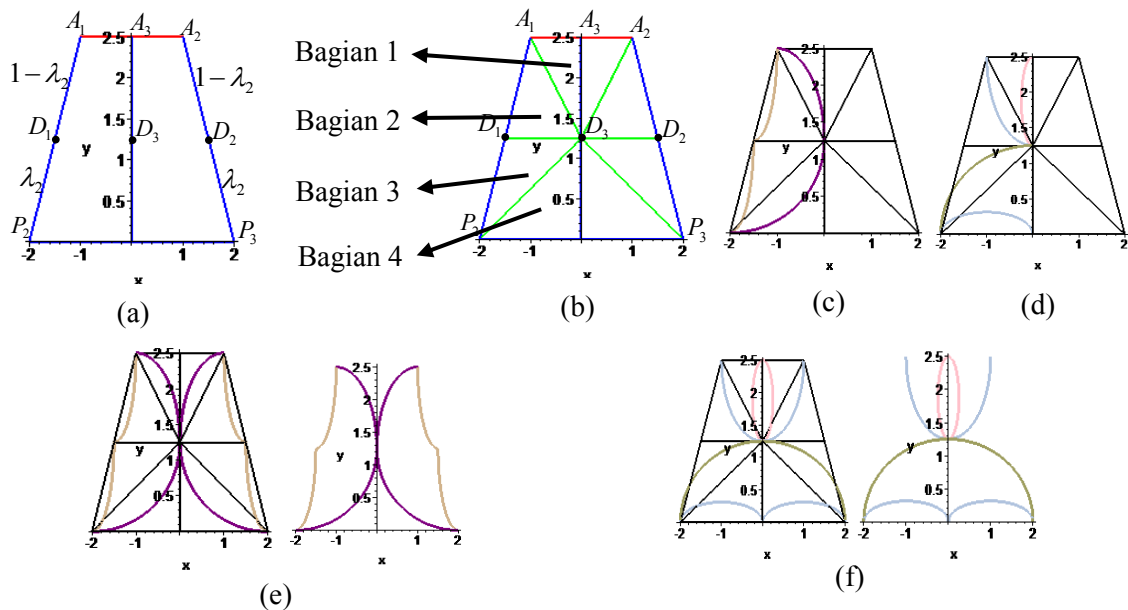
Elips cekung ke arah kiri

- Membangun 1/4 lingkaran cekung ke bawah pada bagian 1 dengan posisi awal ketinggian pada titik A_1 dan posisi akhir ketinggian pada titik D_3 berpusat di (x_{A_1}, y_{D_3}) .
- Membangun 1/4 elips cekung ke atas pada bagian 2 dengan posisi awal ketinggian pada titik A_1 dan posisi akhir ketinggian pada titik D_1 berpusat di (x_{D_1}, y_{A_1}) .
- Membangun 1/4 elips cekung ke atas pada bagian 3 dengan posisi awal ketinggian pada titik D_1 dan posisi akhir ketinggian pada titik P_2 berpusat di (x_{P_2}, y_{D_1}) .
- Membangun 1/4 elips cekung ke atas pada bagian 4 dengan posisi awal ketinggian pada titik D_3 dan posisi akhir ketinggian pada titik P_2 berpusat di (x_{P_2}, y_{D_3}) seperti pada Gambar 4.7c.

Elips cekung ke bawah dan ke arah kanan

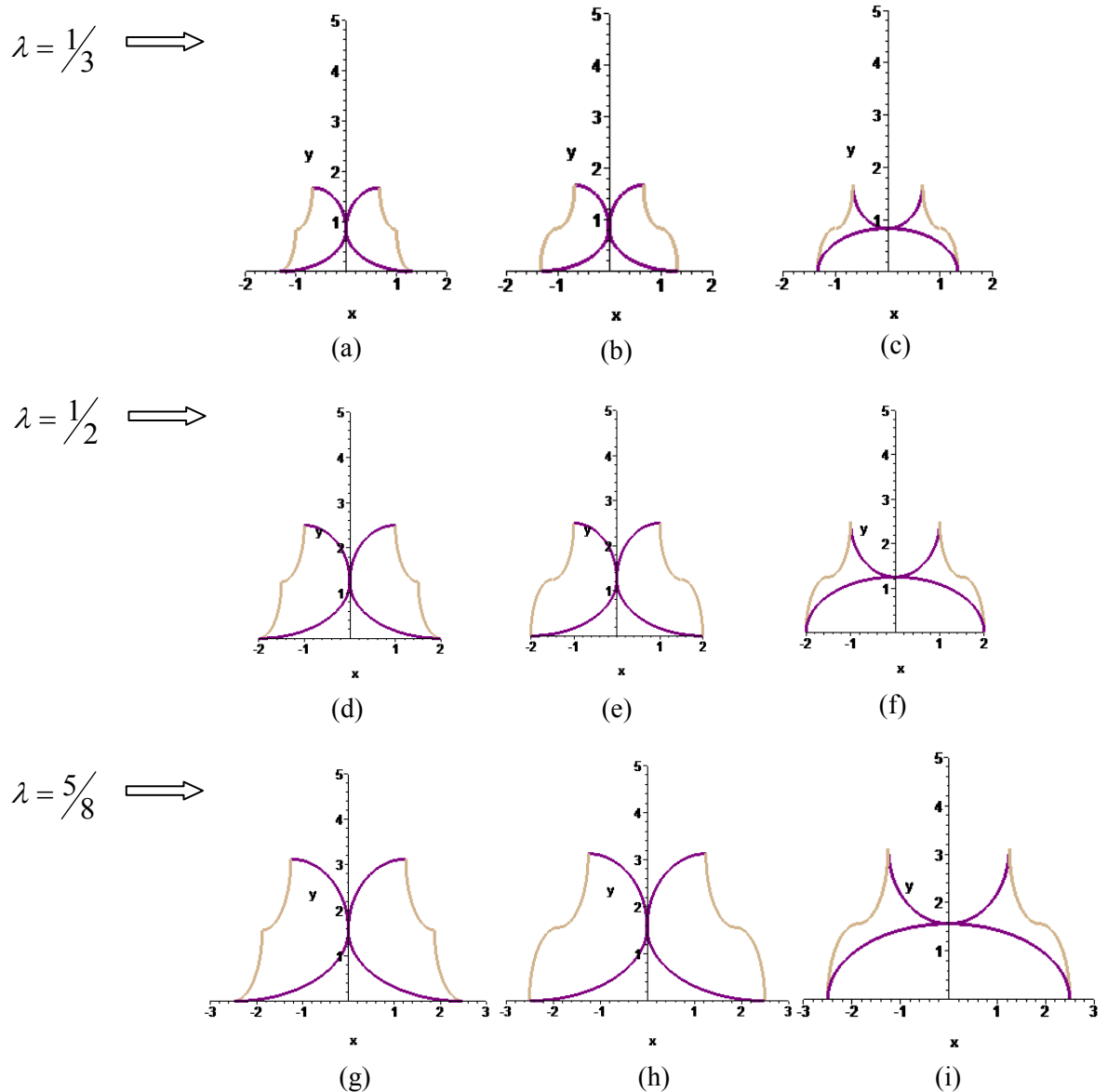
- Membangun 1/2 elips cekung ke arah kanan pada bagian 1 dengan posisi awal ketinggian pada titik A_3 dan posisi akhir ketinggian pada titik D_3 berpusat di $(0, \frac{3}{8}t)$, dengan $2 \leq t \leq 5$ cm.
- Membangun 1/4 lingkaran cekung ke atas pada bagian 2 dengan posisi awal ketinggian pada titik A_1 dan posisi akhir ketinggian pada titik D_3 berpusat di titik A_3 .
- Membangun 1/4 elips cekung ke bawah pada bagian 3 dengan posisi awal ketinggian pada titik P_2 dan posisi akhir ketinggian pada titik D_3 berpusat di titik O .
- Membangun 1/2 elips cekung ke bawah pada bagian 4 dengan posisi awal ketinggian pada titik P_2 dan posisi akhir ketinggian pada titik O berpusat di $(-\frac{1}{2}l, 0)$, dengan $1 \leq l \leq 2$ cm, seperti pada Gambar 4.7d.

5. Refleksikan hasil perlakuan (4) terhadap $\overline{A_3O}$ untuk mengisi tiap bagian daerah cacahan trapesium $A_3OP_3A_2$ (Gambar 4.7e, f).



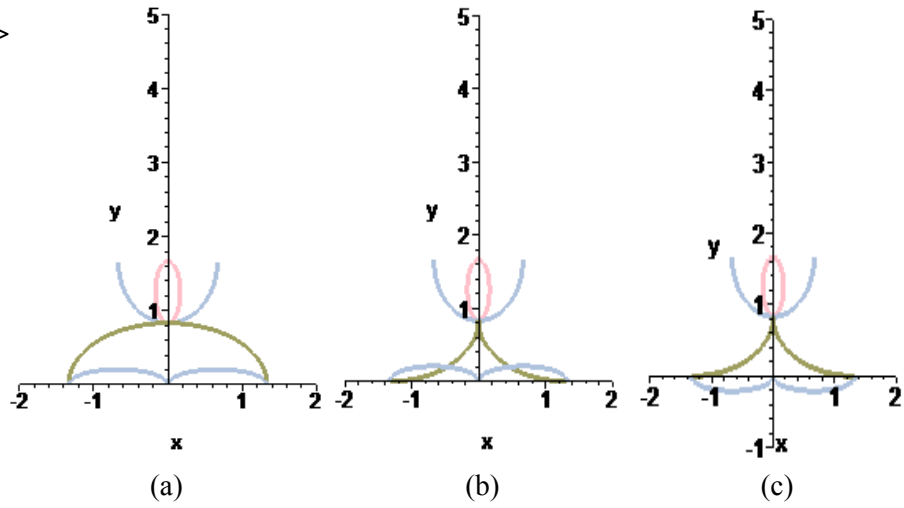
Gambar 4.7 Langkah-langkah memodelisasi liontin pada trapesium $A_1P_2P_3A_2$

Dari Gambar 4.7e dan 4.7f, dapat dilihat perbedaan antara model yang satu dengan model lainnya. Selanjutnya, dapat dikembangkan beberapa bentuk liontin yang bermacam-macam dengan pengambilan nilai λ yang berbeda, yaitu $\lambda = 1/3$, $\lambda = 1/2$, dan $\lambda = 5/8$ yang hasilnya ditunjukkan pada Gambar 4.8 dan Gambar 4.9 dibawah ini.

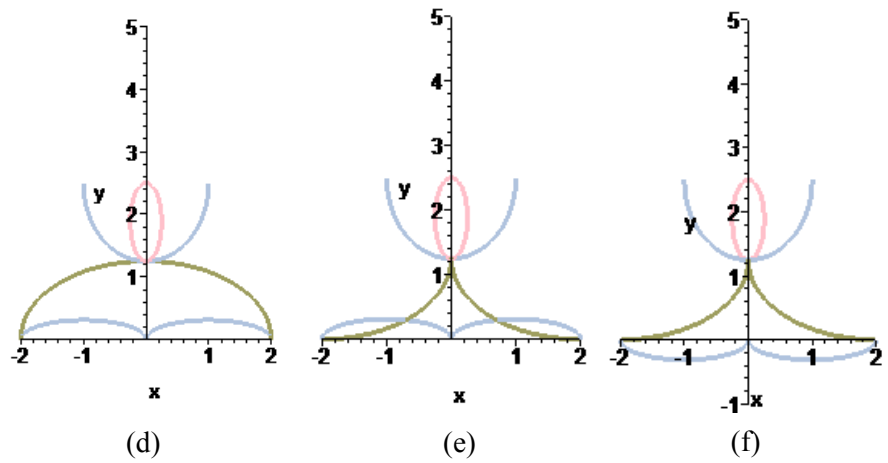


Gambar 4.8 Variasi bentuk liontin elips cekung ke arah kiri
untuk $\lambda = 1/3$, $\lambda = 1/2$, dan $\lambda = 5/8$

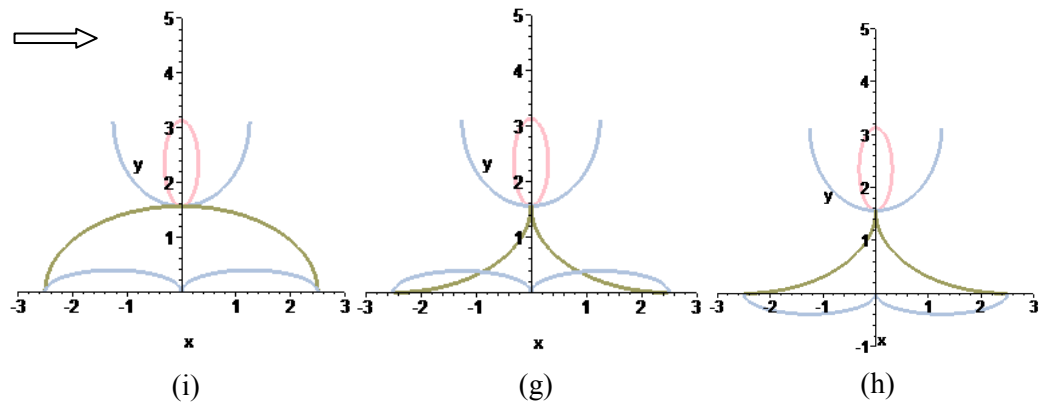
$$\lambda = \frac{1}{3} \implies$$



$$\lambda = \frac{1}{2} \implies$$



$$\lambda = \frac{5}{8} \implies$$



Gambar 4.9 Variasi bentuk liontin elips cekung ke bawah dan ke arah kanan

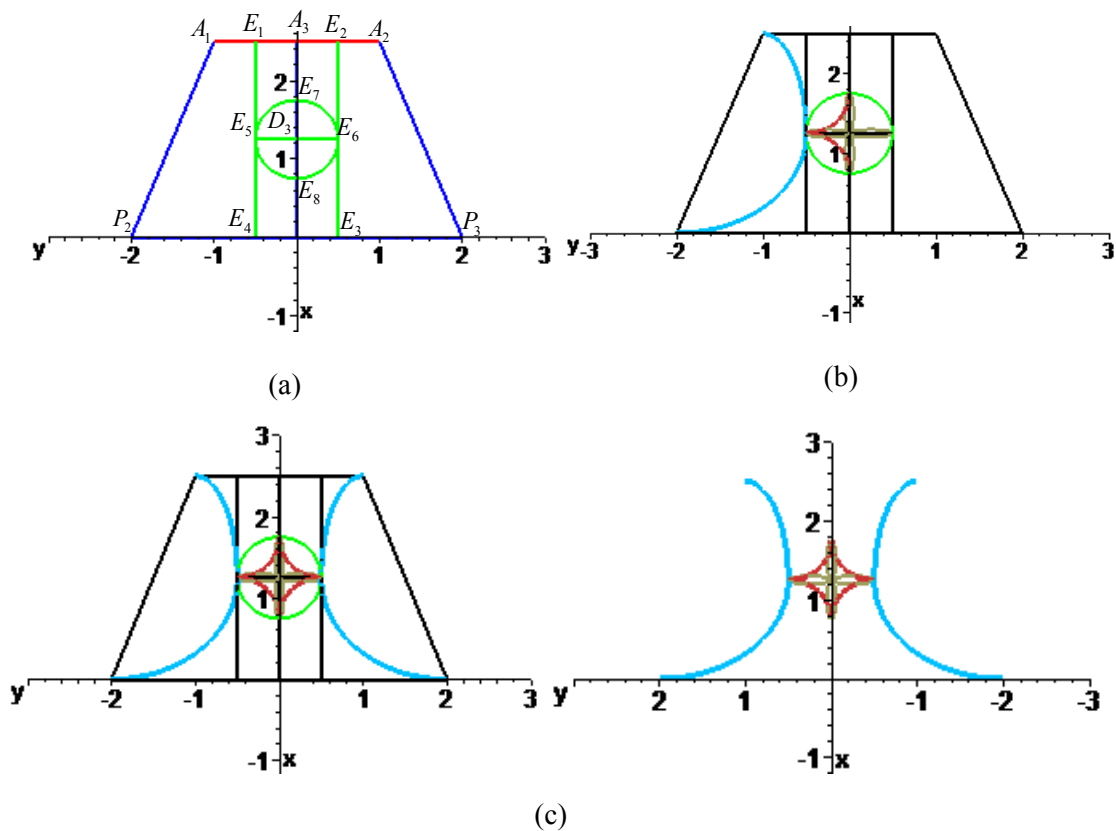
untuk $\lambda = 1/3$, $\lambda = 1/2$, dan $\lambda = 5/8$

4.1.2 Pola Campuran antara Simetri Sumbu dan Simetri Pusat

Prosedur untuk membangun liontin dengan pola campuran simetri sumbu dan simetri pusat sama seperti prosedur pada pola simetri sumbu, dimana pada segitiga $P_1P_2P_3$ terdapat dua bentuk model yang dikembangkan yaitu masing-masing pada segitiga $P_1A_1A_2$ dan trapesium $A_1P_2P_3A_2$. Untuk kasus modelisasi liontin segitiga $P_1A_1A_2$ prosedur yang dikembangkan juga sama seperti pada prosedur pola simetri sumbu. Sedangkan untuk kasus modelisasi liontin trapesium $A_1P_2P_3A_2$ dikembangkan prosedur pola simetri pusat di titik D_3 , yaitu sebagai berikut.

1. Menetapkan posisi titik $E_1(x_{\frac{1}{2}A_1}, y_{A_1})$ dan titik $E_2(x_{\frac{1}{2}A_2}, y_{A_2})$ masing-masing merupakan titik tengah segmen garis $\overline{A_1A_3}$ dan $\overline{A_3A_2}$.
2. Menetapkan posisi titik $E_3(x_{E_1}, y_{P_2})$, titik $E_4(x_{E_2}, y_{P_3})$, titik $E_5(x_{E_1}, y_{D_3})$ dan titik $E_6(x_{E_2}, y_{D_3})$.
3. Membangun segmen garis $\overline{E_1E_3}$, $\overline{E_2E_4}$ dan $\overline{E_5E_6}$ melalui persamaan (2.1).
4. Membangun lingkaran L berjari-jari $r = |\overline{E_5D_3}|$ berpusat di titik D_3 .
5. Menetapkan posisi titik E_7 dan titik E_8 menggunakan formula (2.1), yaitu $\overrightarrow{OE_7} = \lambda_2\overrightarrow{OA_3} + (1 - \lambda_2)\overrightarrow{OD_3}$ dan $\overrightarrow{OE_8} = \lambda_2\overrightarrow{OD_3}$ (Gambar 4.10a).
6. Mengisi tiap bagian dengan bentuk potongan ellips dan potongan lingkaran dengan tahapan sebagai berikut.
 - i. Membangun 4 buah elips pada lingkaran L masing-masing berpusat di $(0, \frac{7}{20}t)$, $(0, \frac{3}{20}t)$, $(-\frac{1}{8}l, \frac{1}{4}t)$, dan $(\frac{1}{8}l, \frac{1}{4}t)$, dengan $2 \leq t \leq 5$ cm.
 - ii. Membangun 1/4 elips cekung ke bawah pada trapesium $A_1P_2E_3E_1$ dengan posisi awal ketinggian pada titik A_1 dan posisi akhir ketinggian pada titik E_5 berpusat di (x_{A_1}, y_{E_5}) .
 - iii. Membangun 1/4 elips cekung ke atas pada trapesium $A_1P_2E_3E_1$ dengan posisi awal ketinggian pada titik E_5 dan posisi akhir ketinggian pada titik P_2 berpusat di (x_{P_2}, y_{E_5}) .

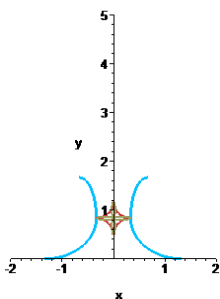
- iv. Membangun 1/4 lingkaran cekung ke atas pada persegi panjang $A_3E_1E_5D_3$ dengan posisi awal ketinggian pada titik E_7 dan posisi akhir ketinggian pada titik E_5 berpusat di (x_{E_5}, y_{E_7}) .
- v. Membangun 1/4 lingkaran cekung ke atas pada persegi panjang $D_3E_5E_3O$ dengan posisi awal ketinggian pada titik E_5 dan posisi akhir ketinggian pada titik E_8 berpusat di (x_{E_5}, y_{E_8}) seperti pada Gambar 4.10b.
- vi. Refleksikan hasil perlakuan (ii-v) terhadap $\overline{A_3O}$ (Gambar 4.10c).



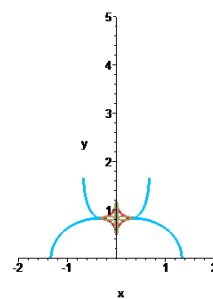
Gambar 4.10 Langkah-langkah memodelisasi liontin pada trapesium $A_1P_2P_3A_2$ dengan pola simetri pusat

Dari Gambar 4.10c, dapat dilihat perbedaan antara model yang satu dengan model lainnya. Selanjutnya, dapat dikembangkan beberapa bentuk liontin yang bermacam-macam dengan pengambilan nilai λ yang berbeda, yaitu $\lambda = 1/3$, $\lambda = 1/2$, dan $\lambda = 5/8$ yang hasilnya ditunjukkan pada Gambar 4.11 dibawah ini.

$$\lambda = \frac{1}{3} \implies$$

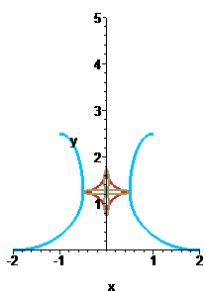


(a)

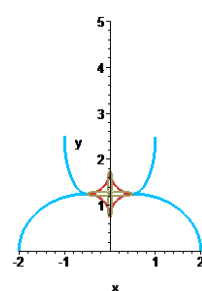


(b)

$$\lambda = \frac{1}{2} \implies$$

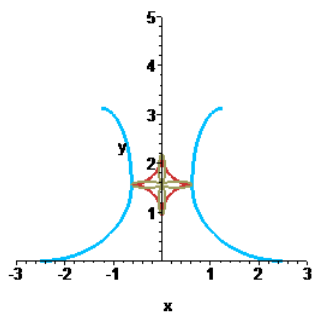


(c)

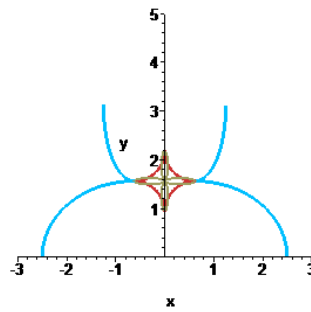


(d)

$$\lambda = \frac{5}{8} \implies$$



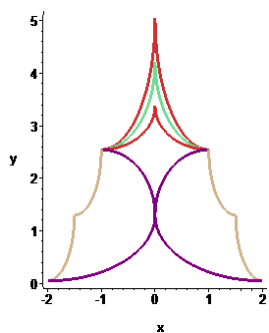
(e)



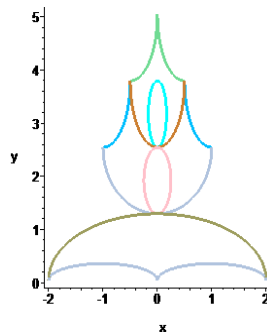
(f)

Gambar 4.11 Variasi bentuk liontin menggunakan pola simetri pusat

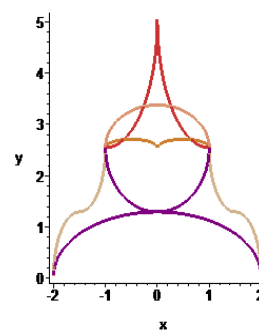
untuk $\lambda = 1/3$, $\lambda = 1/2$, dan $\lambda = 5/8$



(a)



(b)



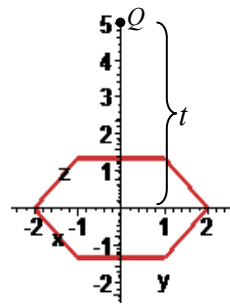
(c)

Gambar 4.12 Variasi model liontin dengan bentuk dasar segitiga sama kaki

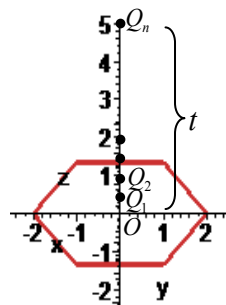
4.2 Modelisasi Kerangka Lontin Cekung dan Cembung

Diberikan segi enam beraturan \mathbb{P}_6 berpusat di $O(0,0,0)$ dengan titik sudut-titik sudutnya $P_1, P_2, P_3, P_4, P_5,$ dan P_6 pada bidang XOY , sedangkan a adalah jarak titik berat O ke titik sudut poligon, dengan $1 \leq a \leq 2$ cm, dan t merupakan tinggi lontin, dengan $3 \leq t \leq 5$ cm (Gambar 4.13a). Berdasarkan data tersebut, ada dua bentuk model yang dikembangkan yaitu kerangka lontin cekung dan kerangka lontin cembung. Prosedur untuk membangun kerangka lontin cekung dan cembung sebagai berikut.

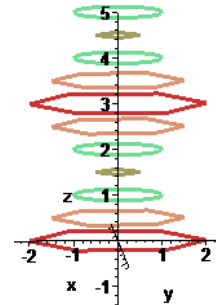
1. Menetapkan posisi titik $Q(0,0,t)$ pada sumbu OZ sehingga $|\overline{OQ}| = t$ (Gambar 4.13a).
2. Membagi \overline{OQ} menjadi n bagian homogen dengan perbandingan ketinggian setiap bagiannya $t_0:t_1:t_2:\dots:t_n$, sehingga terdapat titik $O(0,0,t_0), Q_1(0,0,t_1), Q_2(0,0,t_2), \dots, Q_n(0,0,t_n)$ pada sumbu OZ secara terurut, dengan $Q_n = Q$ (Gambar 4.13b). Dalam hal ini n dipilih dengan batas $6 \leq n \leq 10$, pemilihan batas n tersebut atas dasar pembagian tinggi $(t/0,5)$.
3. Membangun benda-benda geometri datar (poligon segi enam beraturan, lingkaran, dan belah ketupat) pada masing-masing titik O, Q_1, Q_2, \dots, Q_n dengan ketentuan sebagai berikut (Gambar 4.13c):
 - untuk keratan lontin cekung secara terurut pemilihan a dari panjang-pendek-panjang, sedangkan keratan lontin cembung secara terurut pemilihan a dari pendek-panjang-pendek;
 - untuk kesimetrian keratan lontin cekung dan keratan lontin cembung memiliki ketentuan yang sama, yaitu tumpukan bentuk komponen kerangka lontin terstruktur berulang terbalik terhadap komponen semula yang dibangun.
4. Membangun potongan-potongan kurva (segmen garis, lingkaran, dan elips) dari benda geometri datar yang satu ke benda geometri datar berikutnya (Gambar 4.13 d, e).



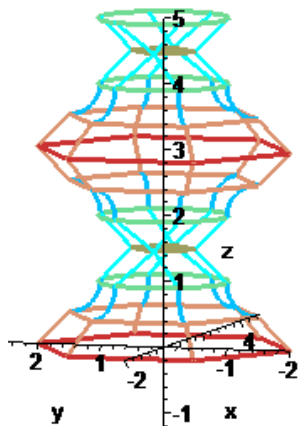
(a)



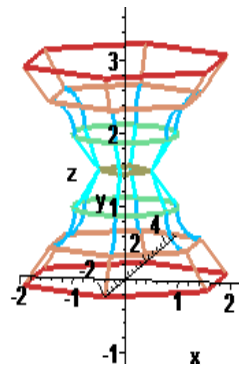
(b)



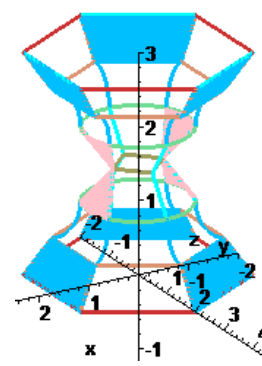
(c)



(1) $n=10$

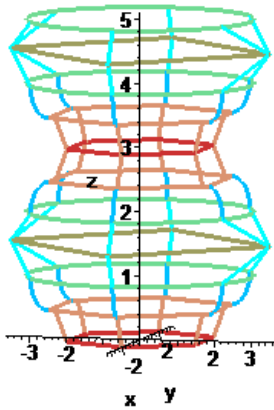


(2) $n=6$

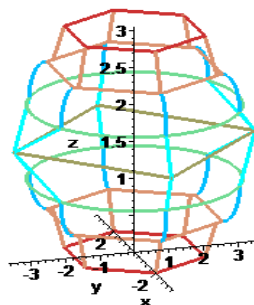


(3) interpolasi

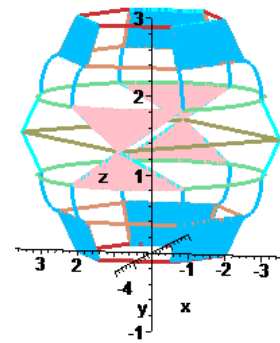
(d) Kerangka liontin cekung



(1) $n=10$



(2) $n=6$



(3) interpolasi

(e) Kerangka liontin cembung

Gambar 4.13 Contoh langkah-langkah modelisasi liontin dengan bentuk dasar poligon segi enam beraturan

4.3 Pembahasan

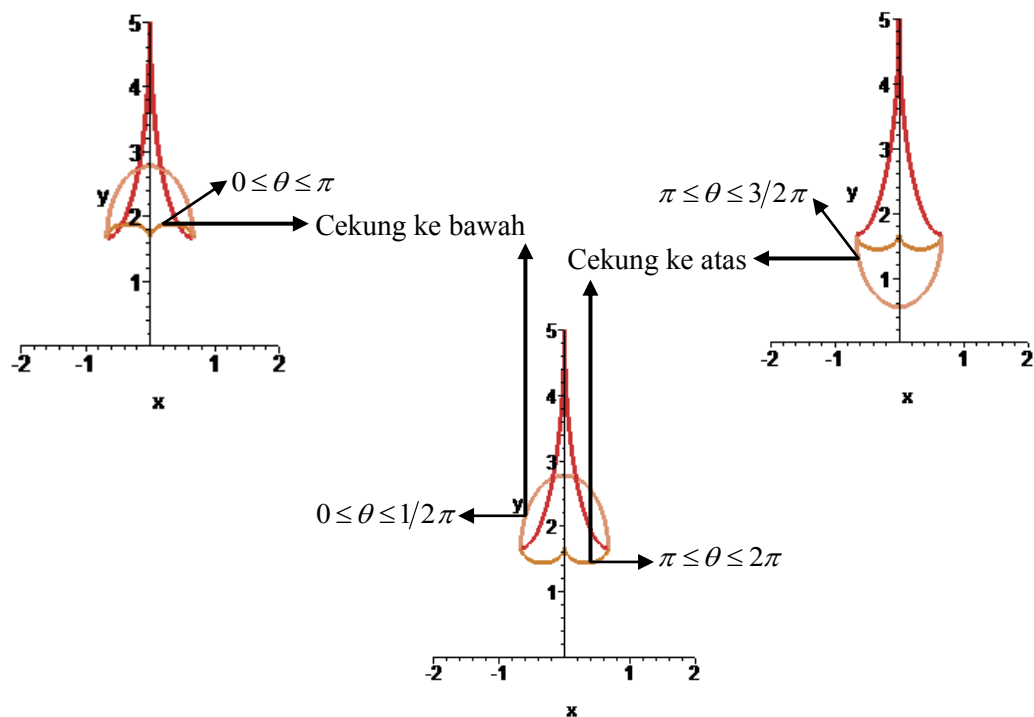
Pada bagian ini dibahas mengenai evaluasi prosedur modelisasi liontin pada dimensi dua dengan bentuk dasar segitiga sama kaki dan modelisasi kerangka liontin cekung dan cembung. Masing-masing prosedur desain tersebut telah dibahas pada subbab 4.1 dan 4.2. Uraian detailnya dijelaskan sebagai berikut.

Sehubungan dengan penerapan prosedur modelisasi liontin pada dimensi dua dengan bentuk dasar segitiga sama kaki, dapat dihasilkan model liontin yang bervariasi. Hal ini dibantu dengan adanya beberapa pola yang dimodelisasi seperti pada Gambar 4.2b, 4.5b, 4.7b, dan 4.10a. Selain itu, beberapa kemudahan yang diberikan sebagai berikut.

- a. Dapat untuk membangun pola liontin berbentuk segitiga sama kaki dan trapesium dengan dasar lengkungan melalui penetapan pemilihan nilai λ bervariasi. Contohnya sebagai berikut:
 1. untuk $\lambda = 1/2$, panjang segmen garis $|\overline{A_3C_3}| = |\overline{C_3P_1}|$ sehingga segitiga $P_1C_1C_2$ dan trapesium $C_1A_1A_2C_2$ memiliki tinggi yang sama (Gambar 4.6c);
 2. untuk $\lambda = 1/3$, panjang segmen garis $|\overline{A_3C_3}| \neq |\overline{C_3P_1}|$ sehingga segitiga $P_1C_1C_2$ dan trapesium $C_1A_1A_2C_2$ memiliki tinggi yang tidak sama, yaitu $t_{P_1C_1C_2} > t_{C_1A_1A_2C_2}$. Selain itu, panjang segmen garis $|\overline{A_3P_1}|$ lebih panjang jika dibandingkan dengan panjang segmen garis $|\overline{A_3P_1}|$ pada $\lambda = 1/2$, sedangkan panjang segmen garis $|\overline{A_1A_2}|$ lebih pendek jika dibandingkan dengan panjang segmen garis $|\overline{A_1A_2}|$ pada $\lambda = 1/2$ (Gambar 4.6a);
 3. untuk $\lambda = 5/8$, panjang segmen garis $|\overline{A_3C_3}| \neq |\overline{C_3P_1}|$ sehingga segitiga $P_1C_1C_2$ dan trapesium $C_1A_1A_2C_2$ memiliki tinggi yang tidak sama, yaitu $t_{P_1C_1C_2} < t_{C_1A_1A_2C_2}$. Selain itu, panjang segmen garis $|\overline{A_3P_1}|$ lebih pendek jika dibandingkan dengan panjang segmen garis $|\overline{A_3P_1}|$ pada $\lambda = 1/2$, sedangkan panjang segmen garis $|\overline{A_1A_2}|$ lebih panjang jika dibandingkan dengan panjang segmen garis $|\overline{A_1A_2}|$ pada $\lambda = 1/2$ (Gambar 4.6e).

Dari ketiga nilai λ tersebut, bentuk validasi model liontin yang paling menarik yaitu untuk $\lambda = 5/8$. Hal ini dikarenakan daerah segitiga lebih kecil daripada daerah trapesium.

- b. Memberikan fasilitas untuk membangun tepian liontin yang bervariasi dengan pemilihan interval besar sudut potongan kurva (θ) yang berbeda. Hal ini menunjukkan bahwa untuk pemilihan interval (θ) yang berbeda maka tepian liontin berbentuk cekung ke arah atas atau ke arah bawah (Gambar 4.4a, b, c).

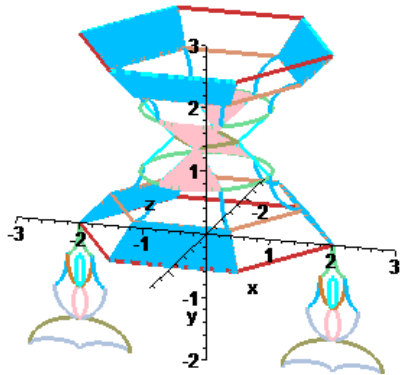


Gambar 4.14 Perubahan cekungan kurva liontin akibat perubahan parameter θ

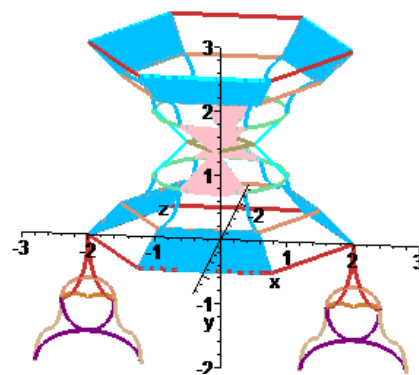
Untuk penerapan prosedur modelisasi kerangka liontin cekung dan cembung, dapat dihasilkan liontin yang beraneka ragam bentuk modelnya. Hal ini dikarenakan penetapan pemilihan nilai a untuk kerangka liontin cekung berbeda dengan kerangka liontin cembung. Selain itu, beberapa keuntungan yang diberikan sebagai berikut.

- a. Dengan memberikan nilai rentang n dalam interval $6 \leq n \leq 10$, kita dapat memberikan variasi ketinggian yang berbeda (Gambar 4.13d1, 4.13d2, 4.13e1, 4.13e2).

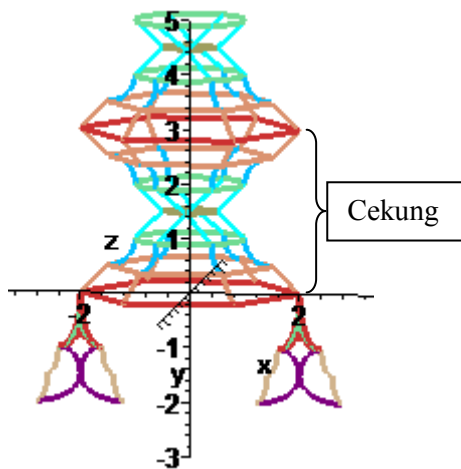
- b. Memberikan fasilitas untuk membangun kerangka liontin dengan penutupan beberapa sisi-sisi yang terbuka menggunakan teknik interpolasi (Gambar 4.13d3, 4.13e3). Hal ini menunjukkan bahwa sisi-sisi terbuka pada kerangka liontin tersebut dapat diisi sehingga membentuk sebuah plat.
- c. Validasi bentuk global dari prosedur 4.1 dan 4.2 dapat dilihat pada contoh Gambar 4.15.



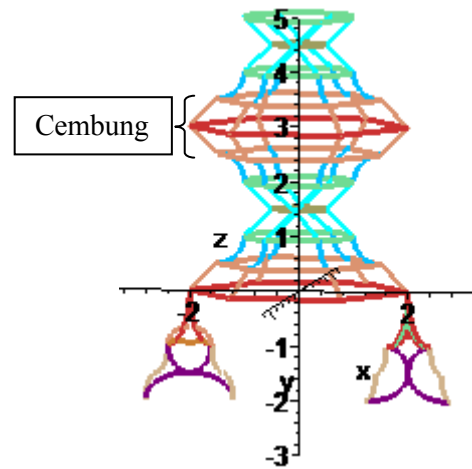
(a) Gabungan antara Gambar 4.13d3 dan 4.12b



(b) Gabungan antara Gambar 4.13d3 dan 4.12c



(c) Gabungan antara Gambar 4.13d1 dan 4.12a



(d) Gabungan antara Gambar 4.13d1, 4.12a, dan 4.12c

4.15 Validasi bentuk global dari prosedur 4.1 dan 4.2

BAB 5. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian dan pembahasan di bab 4, dapat diambil kesimpulan sebagai berikut.

- a. Untuk mendesain beragam bentuk liontin pada dimensi dua dengan bentuk dasar segitiga sama kaki terkomposisi dari potongan kurva (lingkaran atau elips) prosedurnya sebagai berikut.
 1. Menetapkan dua buah titik masing-masing terletak pada sisi kaki segitiga sama kaki dan menarik segmen garis melalui kedua buah titik tersebut sehingga membagi daerah segitiga sama kaki menjadi dua bagian, yaitu daerah segitiga dan daerah trapesium.
 2. Membangun pola-pola bentuk liontin, yaitu:
 - i. pada daerah segitiga
 - membagi ketinggian dan tingkatan segitiga;
 - mengisi tiap tingkatan dengan potongan kurva (lingkaran atau elips) sehingga terbangun pola simetri dan bertingkat;
 - ii. pada daerah trapesium
 - membagi ketinggian dan bagian trapesium;
 - mengisi tiap tingkatan dengan potongan kurva (lingkaran atau elips) sehingga terbangun pola simetri dan bertingkat.
- b. Untuk mendesain beragam bentuk kerangka liontin cekung dan cembung terkomposisi dari belah ketupat, lingkaran, potongan kurva (segmen garis, lingkaran, dan elips), dan keratan limas yang beralaskan poligon segi enam beraturan, prosedurnya sebagai berikut:

1. menetapkan jarak titik berat ke titik sudut alas kerangka liontin serta tinggi kerangka liontin tersebut;
2. menetapkan jumlah dan jenis benda geometri datar dan geometri ruang pembangun kerangka liontin;
3. membangun potongan-potongan kurva (segmen garis, lingkaran, dan elips) pada bidang XOZ dan YOZ dengan titik ujung-titik ujungnya terletak pada titik sudut benda-benda geometri datar dan geometri ruang.

5.2 Saran

Pada skripsi ini telah diperkenalkan prosedur modelisasi liontin kalung dan anting menggunakan benda-benda geometri datar dan geometri ruang seperti segitiga, poligon segi enam beraturan, belah ketupat, lingkaran, elips, dan keratan limas untuk menghasilkan bentuk yang simetris dan bervariasi. Diharapkan untuk penelitian ke depan metode ini dapat dikembangkan lagi dengan menggunakan benda geometri ruang lainnya seperti keratan kerucut, prisma, hiperboloida, dan elipsoida. Selain itu dapat ditawarkan metode konstruksi kurva putar untuk memodelisasi liontin kalung dan anting yang bervariasi dan simetris.

DAFTAR PUSTAKA

- Bastian, A. 2011. *Desain Kap Lampu Duduk Melalui Penggabungan Benda-Benda Geometri Ruang*. Skripsi. Jember : Jurusan Matematika Fakultas MIPA Universitas Jember.
- Budiono, M. 2011. *Pemodelan Handle Pintu Simetris Melalui Penggabungan Beberapa Benda Geometri Ruang*. Skripsi. Jember : Jurusan Matematika Fakultas MIPA Universitas Jember.
- Kriswantoro, A. 2010. *Desain Daun Pintu Melalui Hitung Kesimetrian Geometris*. Skripsi. Jember : Jurusan Matematika Fakultas MIPA Universitas Jember.
- Kusno. 2002. *Geometri Rancang Bangun Studi Aljabar Vektor Garis, Lingkaran dan Ellips*. Jember : Jurusan Matematika Fakultas MIPA Universitas Jember.
- Kusno. 2003. *Geometri Rancang Bangun Studi Hiperbola, Parabola, dan Obyek-Obyek Dasar Geometri Ruang*. Jember : Jurusan Matematika Fakultas MIPA Universitas Jember.

LAMPIRAN

Lampiran A. Modelisasi Lontin pada Dimensi Dua dengan Bentuk Dasar Segitiga Sama Kaki

A.1 Pola Simetri Sumbu

```

> t1:=5:
> t2:=1/2*t1:
> u:=4/5*t2:
> a:=plot3d([(1-t)*0+t*0, (1-t)*0+t*t1, (1-t)*0+t*0],
             t=0..1, v=0..1, color=blue, thickness=2):
> b:=plot3d([(1-t)*u+t*0, (1-t)*0+t*t1, (1-t)*0+t*0],
             t=0..1, v=0..1, color=blue, thickness=2):
> c:=plot3d([(1-t)*(-u)+t*0, (1-t)*0+t*t1, (1-t)*0+t*0],
             t=0..1, v=0..1, color=blue, thickness=2):
> d:=plot3d([(1-t)*u+t*(-u), 0, (1-t)*0+t*0],
             t=0..1, v=0..1, color=blue, thickness=2):
> d1:=plot3d([(1-t)*(1/2*u)+t*(1/2*(-u)), t2, (1-t)*0+t*0],
             t=0..1, v=0..1, color=red, thickness=2):

```

Kasus modelisasi lontin pada segitiga $P_1A_1A_2$

```

> a1:=plot3d([(1-t)*0+t*0, (1-t)*t2+t*t1, (1-t)*0+t*0],
             t=0..1, v=0..1, color=blue, thickness=2):
> b1:=plot3d([(1-t)*(1/2*u)+t*0, (1-t)*t2+t*t1, (1-t)*0+t*0],
             t=0..1, v=0..1, color=blue, thickness=2):
> c1:=plot3d([(1-t)*(1/2*(-u))+t*0, (1-t)*t2+t*t1, (1-t)*0+t*0],
             t=0..1, v=0..1, color=blue, thickness=2):
> d1:=plot3d([(1-t)*(1/2*u)+t*(1/2*(-u)), t2, (1-t)*0+t*0],
             t=0..1, v=0..1, color=red, thickness=2):

```

Pola trap segitiga sama kaki

```

> e1:=plot3d([(1-t)*(1/2*u)+t*0, (1-t)*t2+t*((1/3*t2)+t2),
             (1-t)*0+t*0], t=0..1, v=0..1, color=green,
             thickness=2):
> e2:=plot3d([(1-t)*(1/2*(-u))+t*0, (1-t)*t2+t*((1/3*t2)+t2),
             (1-t)*0+t*0], t=0..1, v=0..1, color=green,
             thickness=2):
> e3:=plot3d([(1-t)*(1/2*u)+t*0, (1-t)*t2+t*((2/3*t2)+t2),
             (1-t)*0+t*0], t=0..1, v=0..1, color=green,
             thickness=2):
> e4:=plot3d([(1-t)*(1/2*(-u))+t*0, (1-t)*t2+t*((2/3*t2)+t2),
             (1-t)*0+t*0], t=0..1, v=0..1, color=green,
             thickness=2):

```

Elips cekung ke atas

```

> f1:=spacecurve([(1/2*(-u))*cos(t)+(1/2*(-u)),(1/3*t2)*sin(t)
+((1/3*t2)+t2),0,t=Pi..3/2*Pi],color=orange,
thickness=3):
> f2:=spacecurve([(1/2*(u))*cos(t)+(1/2*(u)),(1/3*t2)*sin(t) +
((1/3*t2)+t2),0,t=Pi..3/2*Pi],color=orange,
thickness=3):
> f3:=spacecurve([(1/2*(-u))*cos(t)+(1/2*(-u)),
(2/3*t2)*sin(t)+((2/3*t2)+t2),0,t=Pi..3/2*Pi],
color=aquamarine,thickness=3):
> f4:=spacecurve([(1/2*(u))*cos(t)+(1/2*(u)), (2/3*t2)*sin(t)+
((2/3*t2)+t2),0,t=Pi..3/2*Pi],
color=aquamarine,thickness=3):
> f5:=spacecurve([(1/2*(-u))*cos(t)+(1/2*(-u)),(t1-t2)*sin(t)+
t1,0,t=Pi..3/2*Pi],color=orange,thickness=3):
> f6:=spacecurve([(1/2*(u))*cos(t)+(1/2*(u)),(t1-t2)*sin(t)+
t1,0,t=Pi..3/2*Pi],color=orange,thickness=3):

```

Elips cekung ke bawah

```

> f5:=spacecurve([(1/2*(-u))*cos(t)+(1/2*(-u)),(t1-t2)*sin(t)+
t1,0,t=Pi..3/2*Pi],color=orange,thickness=3):
> f6:=spacecurve([(1/2*(u))*cos(t)+(1/2*(u)),(t1-t2)*sin(t)+
t1,0,t=Pi..3/2*Pi],color=orange,thickness=3):
> g1:=spacecurve([(1/4*(-u))*cos(t)+(1/4*(-u)),(1/15*(t1-t2))*
sin(t)+t2,0,t=0..Pi],color=gold,thickness=3):
> g2:=spacecurve([(1/4*(u))*cos(t)+(1/4*(u)),(1/15*(t1-t2))*
sin(t)+t2,0,t=0..Pi],color=gold,thickness=3):
> g3:=spacecurve([(1/2*(-u))*cos(t)+0,(1/3*(t1-t2))*sin(t)+t2,
0,t=0..1/2*Pi],color=tan,thickness=3):
> g4:=spacecurve([(1/2*(u))*cos(t)+0,(1/3*(t1-t2))*sin(t)+t2,
0,t=0..1/2*Pi],color=tan,thickness=3):

```

Pola trap segitiga siku-siku

```

> t1:=5:
> t2:=1/2*t1:
> u:=4/5*t2:
> t3:=1/2*t2:
> a1:=plot3d([(1-t)*0+t*0,(1-t)*t2+t*t1,(1-t)*0+t*0], t=0..1,
v=0..1,thickness=2):
> b1:=plot3d([(1-t)*(1/2*u)+t*0,(1-t)*t2+t*t1,(1-t)*0+t*0],
t=0..1,v=0..1,thickness=2):
> c1:=plot3d([(1-t)*(1/2*(-u))+t*0,(1-t)*t2+t*t1,(1-t)*0+t*0],
t=0..1,v=0..1,thickness=2):
> d1:=plot3d([(1-t)*(1/2*u)+t*(1/2*(-u)),t2,(1-t)*0+t*0],
t=0..1,v=0..1,thickness=2):
> h1:=plot3d([(1-t)*(1/4*u)+t*(-1/4*u)),3/2*t2,(1-t)*0+t*0],
t=0..1,v=0..1,thickness=2):

```

```

> h2:=plot3d([(1-t)*(-1/4*u)+t*0,(1-t)*(t3+t2)+t*t2,(1-t)*0+t*0],t=0..1,v=0..1,thickness=2):
> h3:=plot3d([(1-t)*(1/4*u)+t*0,(1-t)*(t3+t2)+t*t2,(1-t)*0+t*0],t=0..1,v=0..1,thickness=2):
> h4:=plot3d([1/4*u,(1-t)*(t3+t2)+t*t2,(1-t)*0+t*0],t=0..1,v=0..1,thickness=2):
> h5:=plot3d([(-1/4*u),(1-t)*(t3+t2)+t*t2,(1-t)*0+t*0],t=0..1,v=0..1,thickness=2):
> i1:=spacecurve([( (-1/4*u) ) *cos(t)+(-1/4*u) , (t1-(t3+t2)) *sin(t)+t1,0,t=Pi..3/2*Pi],color=aquamarine,thickness=3):
> i2:=spacecurve([(1/4*u) *cos(t)+(1/4*u) , (t1-(t3+t2)) *sin(t)+t1,0,t=Pi..3/2*Pi],color=aquamarine,thickness=3):
> i3:=spacecurve([( (-1/4*u) ) *cos(t)+(-1/2*u) , t3*sin(t)+(t3+t2),0,t=Pi..3/2*Pi],color="DeepSkyBlue",thickness=3):
> i4:=spacecurve([(1/4*u) *cos(t)+(1/2*u) , t3*sin(t)+(t3+t2),0,t=Pi..3/2*Pi],color="DeepSkyBlue",thickness=3):
> i5:=spacecurve([( (-1/4*u) ) *cos(t) , t3*sin(t)+(t3+t2),0,t=Pi..3/2*Pi],color=gold,thickness=3):
> i6:=spacecurve([(1/4*u) *cos(t) , t3*sin(t)+(t3+t2),0,t=Pi..3/2*Pi],color=gold,thickness=3):
> i7:=spacecurve([( (-1/12*u) ) *cos(t) , (1/2*t3) *sin(t)+(t2+(1/2*t3)) , 0,t=3/2*Pi..5/2*Pi],color=cyan,thickness=3):
> i8:=spacecurve([(1/12*u) *cos(t) , (1/2*t3) *sin(t)+(t2+(1/2*t3)) , 0,t=3/2*Pi..5/2*Pi],color=cyan,thickness=3):

```

Kasus modelisasi liontin pada trapesium $A_1P_2P_3A_2$

```

> t1:=5:
> t2:=1/2*t1:
> u:=4/5*t2:
> t3:=1/2*t2:
> a:=plot3d([(1-t)*0+t*0,(1-t)*0+t*t1,(1-t)*0+t*0],t=0..1,v=0..1,color=blue,thickness=2):
> b:=plot3d([(1-t)*u+t*0,(1-t)*0+t*t1,(1-t)*0+t*0],t=0..1,v=0..1,color=blue,thickness=2):
> c:=plot3d([(1-t)*(-u)+t*0,(1-t)*0+t*t1,(1-t)*0+t*0],t=0..1,v=0..1,color=blue,thickness=2):
> d:=plot3d([(1-t)*u+t*(-u),0,(1-t)*0+t*0],t=0..1,v=0..1,color=blue,thickness=2):
> a2:=plot3d([(1-t)*0+t*0,(1-t)*0+t*t2,(1-t)*0+t*0],t=0..1,v=0..1,color=blue,thickness=2):

```

```

> b2:=plot3d([(1-t)*(1/2*u)+t*u, (1-t)*t2+t*0, (1-t)*0+t*0],
             t=0..1, v=0..1, color=blue, thickness=2):
> c2:=plot3d([(1-t)*(1/2*(-u))+t*(-u), (1-t)*t2+t*0, (1-t)*0+
             t*0], t=0..1, v=0..1, color=blue, thickness=2):
> d2:=plot3d([(1-t)*(1/2*u)+t*(1/2*(-u)), t2, (1-t)*0+t*0],
             t=0..1, v=0..1, color=red, thickness=2):

```

Elips cekung ke arah kiri

```

> a3:=plot3d([(1-t)*(-3/4*u)+t*(3/4*u), 1/2*t2, (1-t)*0+t*0],
             t=0..1, v=0..1, color=green, thickness=2):
> b3:=plot3d([(1-t)*0+t*(-1/2*u), (1-t)*(1/2*t2)+t*t2, (1-t)*0+
             t*0], t=0..1, v=0..1, color=green, thickness=2):
> c3:=plot3d([(1-t)*0+t*(1/2*u), (1-t)*(1/2*t2)+t*t2, (1-t)*0+
             t*0], t=0..1, v=0..1, color=green, thickness=2):
> d3:=plot3d([(1-t)*(-u)+t*0, (1-t)*0+t*(1/2*t2), (1-t)*0+t*0],
             t=0..1, v=0..1, color=green, thickness=2):
> e3:=plot3d([(1-t)*(u)+t*0, (1-t)*0+t*(1/2*t2), (1-t)*0+t*0],
             t=0..1, v=0..1, color=green, thickness=2):
> j1:=spacecurve([(-1/2*u)*cos(t)+(-1/2*u), (1/2*t2)*
                 sin(t)+(1/2*t2), 0, t=1/2*Pi..Pi],
                 color="Purple", thickness=3):
> j2:=spacecurve([(1/2*u)*cos(t)+(1/2*u), (1/2*t2)*sin(t)+
                 (1/2*t2), 0, t=1/2*Pi..Pi], color="Purple",
                 thickness=3):
> j3:=spacecurve([(-1/4*u)*cos(t)+(-3/4*u), (1/2*t2)*
                 sin(t)+t2, 0, t=Pi..3/2*Pi], color="Tan",
                 thickness=3):
> j4:=spacecurve([(1/4*u)*cos(t)+(3/4*u), (1/2*t2)*sin(t)+
                 t2, 0, t=Pi..3/2*Pi], color="Tan", thickness=3):
> j5:=spacecurve([(-1/4*u)*cos(t)+(-u), (1/2*t2)*sin(t)+
                 (1/2*t2), 0, t=Pi..3/2*Pi], color="Tan",
                 thickness=3):
> j6:=spacecurve([(1/4*u)*cos(t)+u, (1/2*t2)*sin(t)+(1/2*t2),
                 0, t=Pi..3/2*Pi], color="Tan", thickness=3):
> j7:=spacecurve([(-u)*cos(t)+(-u), (1/2*t2)*sin(t)+(1/2*t2), 0,
                 t=Pi..3/2*Pi], color="Purple", thickness=3):
> j8:=spacecurve([u*cos(t)+u, (1/2*t2)*sin(t)+(1/2*t2), 0,
                 t=Pi..3/2*Pi], color="Purple", thickness=3):

```

Elips cekung ke bawah dan ke arah kanan

```

> k1:=spacecurve([(-1/8*u)*cos(t)+0, (1/4*t2)*sin(t)+
                 (3/4*t2), 0, t=3/2*Pi..5/2*Pi], color=pink,
                 thickness=3):
> k2:=spacecurve([(1/8*u)*cos(t)+0, (1/4*t2)*sin(t)+(3/4*t2),
                 0, t=3/2*Pi..5/2*Pi], color=pink, thickness=3):

```



```

> k3:=spacecurve([(-1/2*u))*cos(t)+0,(1/2*t2)*sin(t)+t2,0,
                 t=3/2*Pi..2*Pi],color="LightSteelBlue",
                 thickness=3):
> k4:=spacecurve([(1/2*u))*cos(t)+0,(1/2*t2)*sin(t)+t2,0,
                 t=3/2*Pi..2*Pi],color="LightSteelBlue",
                 thickness=3):
> k5:=spacecurve([(-u)*cos(t)+0,(1/2*t2)*sin(t)+0,0,
                 t=0..1/2*Pi],color=khaki,thickness=3):
> k6:=spacecurve([u*cos(t)+0,(1/2*t2)*sin(t)+0,0,t=0..1/2*Pi],
                 color=khaki,thickness=3):
> k7:=spacecurve([(-1/2*u))*cos(t)+(-1/2*u),(1/8*t2)*sin(t)+
                 0,0,t=0..Pi],color="LightSteelBlue",
                 thickness=3):
> k8:=spacecurve([(1/2*u))*cos(t)+(1/2*u),(1/8*t2)*sin(t)+0,
                 0,t=0..Pi],color="LightSteelBlue",
                 thickness=3):

```

A.2 Pola Campuran antara Simetri Sumbu dan Simetri Pusat

```

> t1:=5:
> t2:=1/2*t1:
> u:=4/5*t2:
> t3:=1/2*t2:
> a:=plot3d([(1-t)*0+t*0,(1-t)*0+t*t1,(1-t)*0+t*0],
            t=0..1,v=0..1,color=blue,thickness=2):
> b:=plot3d([(1-t)*u+t*0,(1-t)*0+t*t1,(1-t)*0+t*0],
            t=0..1,v=0..1,color=blue,thickness=2):
> c:=plot3d([(1-t)*(-u)+t*0,(1-t)*0+t*t1,(1-t)*0+t*0],
            t=0..1,v=0..1,color=blue,thickness=2):
> d:=plot3d([(1-t)*u+t*(-u),0,(1-t)*0+t*0],
            t=0..1,v=0..1,color=blue,thickness=2):
> a2:=plot3d([(1-t)*0+t*0,(1-t)*0+t*t2,(1-t)*0+t*0],
            t=0..1,v=0..1,color=blue,thickness=2):
> b2:=plot3d([(1-t)*(1/2*u)+t*u,(1-t)*t2+t*0,(1-t)*0+t*0],
            t=0..1,v=0..1,color=blue,thickness=2):
> c2:=plot3d([(1-t)*(1/2*(-u))+t*(-u),(1-t)*t2+t*0,(1-t)*0+
            t*0],t=0..1,v=0..1,color=blue,thickness=2):
> d2:=plot3d([(1-t)*(1/2*u)+t*(1/2*(-u)),t2,(1-t)*0+t*0],
            t=0..1,v=0..1,color=red,thickness=2):
> a4:=plot3d([(1/4*u),(1-t)*0+t*t2,(1-t)*0+t*0],
            t=0..1,v=0..1,color=green,thickness=2):
> b4:=plot3d([(-1/4*u),(1-t)*0+t*t2,(1-t)*0+t*0],
            t=0..1,v=0..1,color=green,thickness=2):
> c4:=spacecurve([(1/4*u)*cos(t)+0,(1/5*t2)*sin(t)+(1/2*t2),0,
                 t=0..2*Pi],color=green,thickness=2):
> d4:=plot3d([(1-t)*(-1/4*u)+t*(1/4*u),(1/2*t2),(1-t)*0+t*0],
            t=0..1,v=0..1,color=green,thickness=2):

```

```

> L1:=spacecurve([(1/4*u)*cos(t)+(-1/2*u),(1/2*t2)*sin(t)+
(1/2*t2),0,t=0..1/2*Pi],color="DeepSkyBlue",
thickness=3):
> L2:=spacecurve([(1/4*u)*cos(t)+(1/2*u),(1/2*t2)*sin(t)+
(1/2*t2),0,t=1/2*Pi..Pi],color="DeepSkyBlue",
thickness=3):
> L3:=spacecurve([(1/4*u)*cos(t)+(-1/4*u),(1/5*t2)*sin(t)+
((1/2*t2)+(1/4*u)),0,t=3/2*Pi..2*Pi],
color=orange,thickness=3):
> L4:=spacecurve([(1/4*u)*cos(t)+(1/4*u),(1/5*t2)*sin(t)+
((1/2*t2)+(1/4*u)),0,t=Pi..3/2*Pi],
color=orange,thickness=3):
> L5:=spacecurve([(1/4*u)*cos(t)+(-1/4*u),(1/5*t2)*sin(t)+
((1/2*t2)-(1/4*u)),0,t=0..1/2*Pi],
color=orange,thickness=3):
> L6:=spacecurve([(1/4*u)*cos(t)+(1/4*u),(1/5*t2)*sin(t)+
((1/2*t2)-
(1/4*u)),0,t=1/2*Pi..Pi],color=orange,
thickness=3):
> L7:=spacecurve([(3/4*u)*cos(t)+u,(1/2*t2)*sin(t)+(1/2*t2),0,
t=Pi..3/2*Pi],color="DeepSkyBlue",
thickness=3):
> L8:=spacecurve([(3/4*u)*cos(t)+(-u),(1/2*t2)*sin(t)+
(1/2*t2),0,t=3/2*Pi..2*Pi],
color="DeepSkyBlue",thickness=3):
> L9:=spacecurve([(1/40*u)*cos(t)+0,((1/2*t2)-(3/4*u))*sin(t)+
((1/2*t2)+(1/8*u)),0,t=0..2*Pi],color=khaki,
thickness=2):
> L10:=spacecurve([(1/40*u)*cos(t)+0,((1/2*t2)-(3/4*u))*sin(t)
+((1/2*t2)-(1/8*u)),0,t=0..2*Pi],color=khaki,
thickness=2):
> L11:=spacecurve([(1/2*t2)-(3/4*u))*cos(t)+(1/8*u),(1/40*u)*
sin(t)+(1/2*t2),0,t=0..2*Pi],color=khaki,
thickness=2):
> L12:=spacecurve([(1/2*t2)-(3/4*u))*cos(t)+(-1/8*u),
(1/40*u)*sin(t)+(1/2*t2),0,t=0..2*Pi],
color=khaki,thickness=2):

```

Lampiran B. Modelisasi Kerangka Lontin Cekung dan Cembung

B.1 Kerangka Lontin Cekung

```

> T1:=5:
> A:=2:

```

```

> a5:=plot3d([(1-t)*0+t*(-sqrt(3)), (1-t)*A+t*1, (1-t)*0+t*0],
             t=0..1, v=0..1, color=orange):
> b5:=plot3d([(1-t)*(-sqrt(3))+t*(-sqrt(3)), (1-t)*1+t*(-1),
             (1-t)*0+t*0], t=0..1, v=0..1, color=orange):
> c5:=plot3d([(1-t)*(-sqrt(3))+t*0, (1-t)*(-1)+t*(-2), (1-t)*0+
             t*0], t=0..1, v=0..1, color=orange):
> d5:=plot3d([(1-t)*0+t*sqrt(3), (1-t)*(-2)+t*(-1), (1-t)*0+
             t*0], t=0..1, v=0..1, color=orange):
> e5:=plot3d([(1-t)*sqrt(3)+t*sqrt(3), (1-t)*(-1)+t*1, (1-t)*0+
             t*0], t=0..1, v=0..1, color=orange):
> f5:=plot3d([(1-t)*sqrt(3)+t*0, (1-t)*1+t*2, (1-t)*0+t*0],
             t=0..1, v=0..1, color=orange):
> a6:=plot3d([(1-t)*0+t*(3/4*(-sqrt(3))), (1-t)*(3/4*A)+t*3/4,
             0.5], t=0..1, v=0..1, color=tan):
> b6:=plot3d([(1-t)*(3/4*(-sqrt(3)))+t*(3/4*(-sqrt(3))),
             (1-t)*(3/4)+t*(-3/4), 0.5],
             t=0..1, v=0..1, color=tan):
> c6:=plot3d([(1-t)*0+t*(3/4*sqrt(3)), (1-t)*(3/4*A)+t*3/4,
             0.5], t=0..1, v=0..1, color=tan):
> d6:=plot3d([(1-t)*(3/4*(-sqrt(3)))+t*0, (1-t)*(-3/4)+t*
             (-6/4), 0.5], t=0..1, v=0..1, color=tan):
> e6:=plot3d([(1-t)*(3/4*sqrt(3))+t*0, (1-t)*(-3/4)+t*(-6/4),
             0.5], t=0..1, v=0..1, color=tan):
> f6:=plot3d([(1-t)*(3/4*sqrt(3))+t*(3/4*sqrt(3)), (1-t)*
             (-3/4)+t*3/4, 0.5], t=0..1, v=0..1, color=tan):
> a7:=spacecurve([(1/2*A)*cos(t)+0, (1/2*A)*sin(t)+0, 1,
                 t=0..2*Pi], color=aquamarine):
> a8:=plot3d([(1-t)*0+t*(1/4*A), (1-t)*(1/4*A)+t*0, 1.5],
             t=0..1, v=0..1, color=khaki):
> b8:=plot3d([(1-t)*0+t*(-1/4*A), (1-t)*(1/4*A)+t*0, 1.5],
             t=0..1, v=0..1, color=khaki):
> c8:=plot3d([(1-t)*0+t*(-1/4*A), (1-t)*(-1/4*A)+t*0, 1.5],
             t=0..1, v=0..1, color=khaki):
> d8:=plot3d([(1-t)*0+t*(1/4*A), (1-t)*(-1/4*A)+t*0, 1.5],
             t=0..1, v=0..1, color=khaki):
> a9:=spacecurve([(1/2*A)*cos(t)+0, (1/2*A)*sin(t)+0, 2,
                 t=0..2*Pi], color=aquamarine):
> a10:=plot3d([(1-t)*0+t*(3/4*(-sqrt(3))), (1-t)*(3/4*A)+t*3/4,
              2.5], t=0..1, v=0..1, color=tan):
> b10:=plot3d([(1-t)*(3/4*(-sqrt(3)))+t*(3/4*(-sqrt(3))),
              (1-t)*(3/4)+t*(-
              3/4), 2.5], t=0..1, v=0..1, color=tan):
> c10:=plot3d([(1-t)*0+t*(3/4*sqrt(3)), (1-t)*(3/4*A)+t*3/4,
              2.5], t=0..1, v=0..1, color=tan):
> d10:=plot3d([(1-t)*(3/4*(-sqrt(3)))+t*0, (1-t)*(-3/4)+t*
              (-6/4), 2.5], t=0..1, v=0..1, color=tan):

```

```

> e10:=plot3d([(1-t)*(3/4*sqrt(3))+t*0,(1-t)*(-3/4)+t*
(-6/4),2.5],t=0..1,v=0..1,color=tan):
> f10:=plot3d([(1-t)*(3/4*sqrt(3))+t*(3/4*sqrt(3)),(1-t)*
(-3/4)+t*3/4,2.5],t=0..1,v=0..1,color=tan):
> a11:=plot3d([(1-t)*0+t*(-sqrt(3)),(1-t)*A+t*1,3],
t=0..1,v=0..1,color=orange):
> b11:=plot3d([(1-t)*(-sqrt(3))+t*(-sqrt(3)),(1-t)*1+t*
(-1),3],t=0..1,v=0..1,color=orange):
> c11:=plot3d([(1-t)*(-sqrt(3))+t*0,(1-t)*(-1)+t*(-2),3],
t=0..1,v=0..1,color=orange):
> d11:=plot3d([(1-t)*0+t*sqrt(3),(1-t)*(-2)+t*(-1),3],
t=0..1,v=0..1,color=orange):
> e11:=plot3d([(1-t)*sqrt(3)+t*sqrt(3),(1-t)*(-1)+t*1,3],
t=0..1,v=0..1,color=orange):
> f11:=plot3d([(1-t)*sqrt(3)+t*0,(1-t)*1+t*2,3],
t=0..1,v=0..1,color=orange):
> a12:=plot3d([(1-t)*0+t*(3/4*(-sqrt(3))), (1-t)*(3/4*A)+t*3/4,
3.5],t=0..1,v=0..1,color=tan):
> b12:=plot3d([(1-t)*(3/4*(-sqrt(3)))+t*(3/4*(-sqrt(3))),
(1-t)*(3/4)+t*(-
3/4),3.5],t=0..1,v=0..1,color=tan):
> c12:=plot3d([(1-t)*0+t*(3/4*sqrt(3)),(1-t)*(3/4*A)+t*3/4,
3.5],t=0..1,v=0..1,color=tan):
> d12:=plot3d([(1-t)*(3/4*(-sqrt(3)))+t*0,(1-t)*(-3/4)+t*
(-6/4),3.5],t=0..1,v=0..1,color=tan):
> e12:=plot3d([(1-t)*(3/4*sqrt(3))+t*0,(1-t)*(-3/4)+t*(-6/4),
3.5],t=0..1,v=0..1,color=tan):
> f12:=plot3d([(1-t)*(3/4*sqrt(3))+t*(3/4*sqrt(3)),(1-t)*
(-3/4)+t*3/4,3.5],t=0..1,v=0..1,color=tan):
> a13:=spacecurve([(1/2*A)*cos(t)+0,(1/2*A)*sin(t)+0,4,
t=0..2*Pi],color=aquamarine):
> a14:=plot3d([(1-t)*0+t*(1/4*A),(1-t)*(1/4*A)+t*0,4.5],
t=0..1,v=0..1,color=khaki):
> b14:=plot3d([(1-t)*0+t*(-1/4*A),(1-t)*(1/4*A)+t*0,4.5],
t=0..1,v=0..1,color=khaki):
> c14:=plot3d([(1-t)*0+t*(-1/4*A),(1-t)*(-1/4*A)+t*0,4.5],
t=0..1,v=0..1,color=khaki):
> d14:=plot3d([(1-t)*0+t*(1/4*A),(1-t)*(-1/4*A)+t*0,4.5],
t=0..1,v=0..1,color=khaki):
> a15:=spacecurve([(1/2*A)*cos(t)+0,(1/2*A)*sin(t)+0,5,
t=0..2*Pi],color=aquamarine):
> a16:=plot3d([(1-t)*0+t*0,(1-t)*(-A)+t*(-3/4*A),(1-t)*0+t*
(0.5)],t=0..1,v=0..1,color=tan):
> a17:=plot3d([(1-t)*0+t*0,(1-t)*(A)+t*(3/4*A),(1-t)*0+t*
(0.5)],t=0..1,v=0..1,color=tan):
> a18:=plot3d([(1-t)*sqrt(3)+t*(3/4*sqrt(3)),(1-t)*(-1)+t*
(-3/4),(1-t)*0+t*(0.5)],t=0..1,v=0..1,color=tan):

```

```

> a19:=plot3d([(1-t)*sqrt(3)+t*(3/4*sqrt(3)),(1-t)*(1)+t*(3/4),
(1-t)*0+t*(0.5)],t=0..1,v=0..1,color=tan):
> a20:=plot3d([(1-t)*(-sqrt(3))+t*(-3/4*sqrt(3)),(1-t)*(1)+
t*(3/4),(1-t)*0+t*(0.5)],
t=0..1,v=0..1,color=tan):
> a21:=plot3d([(1-t)*(-sqrt(3))+t*(-3/4*sqrt(3)),(1-t)*(-1)+
t*(-3/4),(1-t)*0+t*(0.5)],
t=0..1,v=0..1,color=tan):
> b15:=spacecurve([0+0*1/2*cos(t)+0*1/2*sin(t),(-3/2)+(-1)*
1/2*cos(t)+0*1/2*sin(t),1+0*1/2*cos(t)+1*1/2
*sin(t),t=Pi..3/2*Pi],color="DeepSkyBlue"):
> b16:=spacecurve([(-3/4*sqrt(3))+(-1/2*sqrt(3))*1/2*cos(t)+0*
1/2*sin(t),(-3/4)+(-1/2)*1/2*cos(t)+0*1/2*
sin(t),1+0*1/2*cos(t)+1*1/2*sin(t),
t=Pi..3/2*Pi],color="DeepSkyBlue"):
> b17:=spacecurve([(-3/4*sqrt(3))+(-1/2*sqrt(3))*1/2*cos(t)+0*
1/2*sin(t),3/4+(1/2)*1/2*cos(t)+0*1/2*sin(t)
,1+0*1/2*cos(t)+1*1/2*sin(t),t=Pi..3/2*Pi],
color="DeepSkyBlue"):
> b18:=spacecurve([0+0*1/2*cos(t)+0*1/2*sin(t),(3/2)+1*1/2*
cos(t)+0*1/2*sin(t),1+0*1/2*cos(t)+1*1/2*
sin(t),t=Pi..3/2*Pi],color="DeepSkyBlue"):
> b19:=spacecurve([(3/4*sqrt(3))+(1/2*sqrt(3))*1/2*cos(t)+0*
1/2*sin(t),(3/4)+(1/2)*1/2*cos(t)+0*1/2*
sin(t),1+0*1/2*cos(t)+1*1/2*sin(t),
t=Pi..3/2*Pi],color="DeepSkyBlue"):
> b20:=spacecurve([(3/4*sqrt(3))+(1/2*sqrt(3))*1/2*cos(t)+0*
1/2*sin(t),(-3/4)+(-1/2)*1/2*cos(t)+0*
1/2*sin(t),1+0*1/2*cos(t)+1*1/2*sin(t),
t=Pi..3/2*Pi],color="DeepSkyBlue"):
> b21:=plot3d([(1-t)*0+t*0,(1-t)*(-1/2*A)+t*(-1/4*A),(1-t)*1+
t*3/2],t=0..1,v=0..1,color=cyan):
> b22:=plot3d([(1-t)*0+t*0,(1-t)*(1/2*A)+t*(1/4*A),(1-t)*1+
t*3/2],t=0..1,v=0..1,color=cyan):
> b23:=plot3d([(1-t)*((-1/2)*sqrt(3))+t*(-1/2),(1-t)*(1/4*A)+
t*0,(1-t)*1+t*3/2],t=0..1,v=0..1,color=cyan):
> b24:=plot3d([(1-t)*((-1/2)*sqrt(3))+t*(-1/2),(1-t)*(-1/4*A)+
t*0,(1-t)*1+t*3/2],t=0..1,v=0..1,color=cyan):
> b25:=plot3d([(1-t)*((1/2)*sqrt(3))+t*(1/2),(1-t)*(-1/4*A)+
t*0,(1-t)*1+t*3/2],t=0..1,v=0..1,color=cyan):
> b26:=plot3d([(1-t)*((1/2)*sqrt(3))+t*(1/2),(1-t)*(1/4*A)+
t*0,(1-t)*1+t*3/2],t=0..1,v=0..1,color=cyan):
> g1:=plot3d([(1-t)*0+t*0,(1-t)*(-1/4*A)+t*(-1/2*A),(1-t)*3/2+
t*2],t=0..1,v=0..1,color=cyan):
> g2:=plot3d([(1-t)*0+t*0,(1-t)*(1/4*A)+t*(1/2*A),(1-t)*3/2+
t*2],t=0..1,v=0..1,color=cyan):

```

```

> g3:=plot3d([(1-t)*(-1/2)+t*((-1/2)*sqrt(3)), (1-t)*0+t*
(1/4*A), (1-t)*3/2+t*2],t=0..1,v=0..1,color=cyan):
> g4:=plot3d([(1-t)*(-1/2)+t*((-1/2)*sqrt(3)), (1-t)*0+t*
(-1/4*A), (1-t)*3/2+t*2],t=0..1,v=0..1,color=cyan):
> g5:=plot3d([(1-t)*(1/2)+t*((1/2)*sqrt(3)), (1-t)*0+t*
(-1/4*A), (1-t)*3/2+t*2],t=0..1,v=0..1,color=cyan):
> g6:=plot3d([(1-t)*(1/2)+t*((1/2)*sqrt(3)), (1-t)*0+t*(1/4*A),
(1-t)*3/2+t*2],t=0..1,v=0..1,color=cyan):
> h1:=spacecurve([0+0*1/2*cos(t)+0*1/2*sin(t), (-3/2)+(-1)*
1/2*cos(t)+0*1/2*sin(t), 2+0*1/2*cos(t)+1*1/2*s
in(t), t=1/2*Pi..Pi],color="DeepSkyBlue"):
> h2:=spacecurve([(-3/4*sqrt(3))+(-1/2*sqrt(3))*1/2*cos(t)+
0*1/2*sin(t), (-3/4)+(-1/2)*1/2*cos(t)+0*
1/2*sin(t), 2+0*1/2* cos(t)+1*1/2*sin(t),
t=1/2*Pi..Pi],color="DeepSkyBlue"):
> h3:=spacecurve([(-3/4*sqrt(3))+(-1/2*sqrt(3))*1/2*cos(t)+0*
1/2*sin(t), 3/4+(1/2)*1/2*cos(t)+0*1/2*sin(t),
2+0*1/2*cos(t)+1*1/2*sin(t),t=1/2*Pi..Pi],
color="DeepSkyBlue"):
> h4:=spacecurve([0+0*1/2*cos(t)+0*1/2*sin(t), (3/2)+1*1/2*
cos(t)+0*1/2*sin(t), 2+0*1/2*cos(t)+1*1/2*
sin(t), t=1/2*Pi..Pi],color="DeepSkyBlue"):
> h5:=spacecurve([(3/4*sqrt(3))+(1/2*sqrt(3))*1/2*cos(t)+
0*1/2*sin(t), (3/4)+(1/2)*1/2*cos(t)+0*1/2*
sin(t), 2+0*1/2* cos(t)+1*1/2*sin(t),
t=1/2*Pi..Pi],color="DeepSkyBlue"):
> h6:=spacecurve([(3/4*sqrt(3))+(1/2*sqrt(3))*1/2*cos(t)+
0*1/2*sin(t), (-3/4)+(-1/2)*1/2*cos(t)+0*
1/2*sin(t), 2+0*1/2* cos(t)+1*1/2*sin(t),
t=1/2*Pi..Pi],color="DeepSkyBlue"):
> c16:=plot3d([(1-t)*0+t*0, (1-t)*(-3/4*A)+t*(-A),
(1-t)*2.5+t*3],t=0..1,v=0..1,color=tan):
> c17:=plot3d([(1-t)*0+t*0, (1-t)*(3/4*A)+t*A,
(1-t)*2.5+t*3],t=0..1,v=0..1,color=tan):
> c18:=plot3d([(1-t)*(3/4*sqrt(3))+t*sqrt(3), (1-t)*(-3/4)+
t*(-1), (1-t)*2.5+t*3],t=0..1,v=0..1,color=tan):
> c19:=plot3d([(1-t)*(3/4*sqrt(3))+t*sqrt(3),
(1-t)*(3/4)+t*1, (1-t)*2.5+t*3],t=0..1,v=0..1,
color=tan):
> c20:=plot3d([(1-t)*(-3/4*sqrt(3))+t*(-sqrt(3)),
(1-t)*(3/4)+t*1, (1-t)*2.5+t*3],t=0..1,v=0..1,
color=tan):
> c21:=plot3d([(1-t)*(-3/4*sqrt(3))+t*(-sqrt(3)), (1-t)*
(-3/4)+t*(-1), (1-t)*2.5+t*3],t=0..1,v=0..1,
color=tan):
> d16:=plot3d([(1-t)*0+t*0, (1-t)*(-A)+t*(-3/4*A),
(1-t)*3+t*(3.5)],t=0..1,v=0..1,color=tan):

```

```

> d17:=plot3d([(1-t)*0+t*0, (1-t)*(A)+t*(3/4*A),
              (1-t)*3+t*(3.5)],t=0..1,v=0..1,color=tan):
> d18:=plot3d([(1-t)*sqrt(3)+t*(3/4*sqrt(3)), (1-t)*(-1)+t*
              (-3/4), (1-t)*3+t*(3.5)],t=0..1,v=0..1,color=tan):
> d19:=plot3d([(1-t)*sqrt(3)+t*(3/4*sqrt(3)),
              (1-t)*(1)+t*(3/4), (1-t)*3+t*(3.5)],t=0..1,v=0..1,
              color=tan):
> d20:=plot3d([(1-t)*(-sqrt(3))+t*(-3/4*sqrt(3)),
              (1-t)*(1)+t*(3/4), (1-t)*3+t*(3.5)],t=0..1,v=0..1,
              color=tan):
> d21:=plot3d([(1-t)*(-sqrt(3))+t*(-3/4*sqrt(3)), (1-t)*
              (-1)+t*(-3/4), (1-t)*3+t*(3.5)],t=0..1,v=0..1,
              color=tan):
> e15:=spacecurve([0+0*1/2*cos(t)+0*1/2*sin(t), (-3/2)+
                  (-1)*1/2*cos(t)+0*1/2*sin(t), 4+0*1/2
                  *cos(t)+1*1/2*sin(t),t=Pi..3/2*Pi],
                  color="DeepSkyBlue"):
> e16:=spacecurve([(3/4*sqrt(3))+(-1/2*sqrt(3))*1/2*cos(t)+
                  0*1/2*sin(t), (-3/4)+(-1/2)*1/2*cos(t)+0*1/2*
                  sin(t), 4+ 0*1/2*cos(t)+1*1/2*sin(t),
                  t=Pi..3/2*Pi],color="DeepSkyBlue"):
> e17:=spacecurve([(3/4*sqrt(3))+(-1/2*sqrt(3))*1/2*cos(t)+
                  0*1/2*sin(t), 3/4+(1/2)*1/2*cos(t)+0*1/2*sin(t)
                  ), 4+0*1/2*cos(t)+1*1/2*sin(t),t=Pi..3/2*Pi],
                  color="DeepSkyBlue"):
> e18:=spacecurve([0+0*1/2*cos(t)+0*1/2*sin(t), (3/2)+1*1/2*
                  cos(t)+0*1/2*sin(t), 4+0*1/2*cos(t)+1*
                  1/2*sin(t),t=Pi..3/2*Pi],
                  color="DeepSkyBlue"):
> e19:=spacecurve([(3/4*sqrt(3))+1/2*sqrt(3))*1/2*cos(t)+
                  0*1/2*sin(t), (3/4)+(1/2)*1/2*cos(t)+0*1/2*sin
                  (t), 4+0*1/2*cos(t)+1*1/2*sin(t),
                  t=Pi..3/2*Pi],color="DeepSkyBlue"):
> e20:=spacecurve([(3/4*sqrt(3))+1/2*sqrt(3))*1/2*cos(t)+
                  0*1/2*sin(t), (-3/4)+(-1/2)*1/2*cos(t)+0*1/2
                  *sin(t), 4+0*1/2*cos(t)+1*1/2*sin(t),
                  t=Pi..3/2*Pi],color="DeepSkyBlue"):
> e21:=plot3d([(1-t)*0+t*0, (1-t)*(-1/2*A)+t*(-1/4*A), (1-t)*4+
              t*9/2],t=0..1,v=0..1,color=cyan):
> e22:=plot3d([(1-t)*0+t*0, (1-t)*(1/2*A)+t*(1/4*A), (1-t)*4+
              t*9/2],t=0..1,v=0..1,color=cyan):
> e23:=plot3d([(1-t)*((-1/2)*sqrt(3))+t*(-1/2), (1-t)*(1/4*A)+
              t*0, (1-t)*4+t*9/2],t=0..1,v=0..1,color=cyan):
> e24:=plot3d([(1-t)*((-1/2)*sqrt(3))+t*(-1/2), (1-t)*(-1/4*A)+
              t*0, (1-t)*4+t*9/2],t=0..1,v=0..1,color=cyan):
> e25:=plot3d([(1-t)*((1/2)*sqrt(3))+t*(1/2), (1-t)*(-1/4*A)+
              t*0, (1-t)*4+t*9/2],t=0..1,v=0..1,color=cyan):

```

```

> e26:=plot3d([(1-t)*((1/2)*sqrt(3))+t*(1/2), (1-t)*(1/4*A)+
t*0, (1-t)*4+t*9/2],t=0..1,v=0..1,color=cyan):
> g7:=plot3d([(1-t)*0+t*0, (1-t)*(-1/4*A)+t*(-1/2*A), (1-t)*9/2+
t*5],t=0..1,v=0..1,color=cyan):
> g8:=plot3d([(1-t)*0+t*0, (1-t)*(1/4*A)+t*(1/2*A), (1-t)*9/2+
t*5],t=0..1,v=0..1,color=cyan):
> g9:=plot3d([(1-t)*(-1/2)+t*((-1/2)*sqrt(3)), (1-t)*0+t*
(1/4*A), (1-t)*9/2+t*5],t=0..1,v=0..1,color=cyan):
> g10:=plot3d([(1-t)*(-1/2)+t*((-1/2)*sqrt(3)), (1-t)*0+t*
(-1/4*A), (1-t)*9/2+t*5],t=0..1,v=0..1,
color=cyan):
> g11:=plot3d([(1-t)*(1/2)+t*((1/2)*sqrt(3)), (1-t)*0+t*
(-1/4*A), (1-t)*9/2+t*5],t=0..1,v=0..1,
color=cyan):
> g12:=plot3d([(1-t)*(1/2)+t*((1/2)*sqrt(3)), (1-t)*0+t*
(1/4*A), (1-t)*9/2+t*5],t=0..1,v=0..1,color=cyan):

```

B.2 Kerangka Lontin Cembung

```

> a5:=plot3d([(1-t)*0+t*(-sqrt(3)), (1-t)*A+t*1, (1-t)*0+t*0],
t=0..1,v=0..1,color=orange):
> b5:=plot3d([(1-t)*(-sqrt(3))+t*(-sqrt(3)), (1-t)*1+t*(-1),
(1-t)*0+t*0],t=0..1,v=0..1,color=orange):
> c5:=plot3d([(1-t)*(-sqrt(3))+t*0, (1-t)*(-1)+t*(-2), (1-t)*0+
t*0],t=0..1,v=0..1,color=orange):
> d5:=plot3d([(1-t)*0+t*sqrt(3), (1-t)*(-2)+t*(-1), (1-t)*0+
t*0],t=0..1,v=0..1,color=orange):
> e5:=plot3d([(1-t)*sqrt(3)+t*sqrt(3), (1-t)*(-1)+t*1, (1-t)*0+
t*0],t=0..1,v=0..1,color=orange):
> f5:=plot3d([(1-t)*sqrt(3)+t*0, (1-t)*1+t*2, (1-t)*0+t*0],
t=0..1,v=0..1,color=orange):
> a6:=plot3d([(1-t)*0+t*(-5/4*(sqrt(3))), (1-t)*(5/2)+t*5/4,
0.5],t=0..1,v=0..1,color=tan):
> b6:=plot3d([(1-t)*(-5/4*(sqrt(3)))+t*(-5/4*(sqrt(3))),
(1-t)*(5/4)+t*(-
5/4),0.5],t=0..1,v=0..1,color=tan):
> c6:=plot3d([(1-t)*(-5/4*sqrt(3))+t*0, (1-t)*(-5/4)+t*(-5/2),
0.5],t=0..1,v=0..1,color=tan):
> d6:=plot3d([(1-t)*0+t*(5/4*(sqrt(3))), (1-t)*(-5/2)+t*
(-5/4),0.5],t=0..1,v=0..1,color=tan):
> e6:=plot3d([(1-t)*(5/4*sqrt(3))+t*(5/4*sqrt(3)), (1-t)*
(-5/4)+t*(5/4),0.5],t=0..1,v=0..1,color=tan):
> f6:=plot3d([(1-t)*(5/4*sqrt(3))+t*0, (1-t)*(5/4)+t*5/2,0.5],
t=0..1,v=0..1,color=tan):
> a7:=spacecurve([(6/4*A)*cos(t)+0, (6/4*A)*sin(t)+0,1,
t=0..2*Pi],color=aquamarine):

```



```

> a8:=plot3d([(1-t)*0+t*(7/4*A), (1-t)*(7/4*A)+t*0,1.5],
             t=0..1,v=0..1,color=khaki):
> b8:=plot3d([(1-t)*0+t*(-7/4*A), (1-t)*(7/4*A)+t*0,1.5],
             t=0..1,v=0..1,color=khaki):
> c8:=plot3d([(1-t)*0+t*(-7/4*A), (1-t)*(-7/4*A)+t*0,1.5],
             t=0..1,v=0..1,color=khaki):
> d8:=plot3d([(1-t)*0+t*(7/4*A), (1-t)*(-7/4*A)+t*0,1.5],
             t=0..1,v=0..1,color=khaki):
> a9:=spacecurve([(6/4*A)*cos(t)+0, (6/4*A)*sin(t)+0,2,
                 t=0..2*Pi],color=aquamarine):
> a10:=plot3d([(1-t)*0+t*(-5/4*(sqrt(3))), (1-t)*(5/2)+t*5/4,
              2.5],t=0..1,v=0..1,color=tan):
> b10:=plot3d([(1-t)*(-5/4*(sqrt(3)))+t*(-5/4*(sqrt(3))),
              (1-t)*(5/4)+t*(-5/4),2.5],t=0..1,v=0..1,
              color=tan):
> c10:=plot3d([(1-t)*(-5/4*sqrt(3))+t*0, (1-t)*(-5/4)+t*(-5/2),
              2.5],t=0..1,v=0..1,color=tan):
> d10:=plot3d([(1-t)*0+t*(5/4*(sqrt(3))), (1-t)*(-5/2)+t*
              (-5/4),2.5],t=0..1,v=0..1,color=tan):
> e10:=plot3d([(1-t)*(5/4*sqrt(3))+t*(5/4*sqrt(3)), (1-t)*
              (-5/4)+t*(5/4),2.5],t=0..1,v=0..1,color=tan):
> f10:=plot3d([(1-t)*(5/4*sqrt(3))+t*0, (1-t)*(5/4)+t*5/2,2.5],
              t=0..1,v=0..1,color=tan):
> a11:=plot3d([(1-t)*0+t*(-sqrt(3)), (1-t)*A+t*1,3],
              t=0..1,v=0..1,color=orange):
> b11:=plot3d([(1-t)*(-sqrt(3))+t*(-sqrt(3)), (1-t)*1+t*
              (-1),3],t=0..1,v=0..1,color=orange):
> c11:=plot3d([(1-t)*(-sqrt(3))+t*0, (1-t)*(-1)+t*(-2),3],
              t=0..1,v=0..1,color=orange):
> d11:=plot3d([(1-t)*0+t*sqrt(3), (1-t)*(-2)+t*(-1),3],
              t=0..1,v=0..1,color=orange):
> e11:=plot3d([(1-t)*sqrt(3)+t*sqrt(3), (1-t)*(-1)+t*1,3],
              t=0..1,v=0..1,color=orange):
> f11:=plot3d([(1-t)*sqrt(3)+t*0, (1-t)*1+t*2,3],
              t=0..1,v=0..1,color=orange):
> a12:=plot3d([(1-t)*0+t*(-5/4*(sqrt(3))), (1-t)*(5/2)+t*5/4,
              3.5],t=0..1,v=0..1,color=tan):
> b12:=plot3d([(1-t)*(-5/4*(sqrt(3)))+t*(-5/4*(sqrt(3))),
              (1-t)*(5/4)+t*(-5/4),3.5],t=0..1,v=0..1,
              color=tan):
> c12:=plot3d([(1-t)*(-5/4*sqrt(3))+t*0, (1-t)*(-5/4)+t*(-5/2),
              3.5],t=0..1,v=0..1,color=tan):
> d12:=plot3d([(1-t)*0+t*(5/4*(sqrt(3))), (1-t)*(-5/2)+t*
              (-5/4),3.5],t=0..1,v=0..1,color=tan):
> e12:=plot3d([(1-t)*(5/4*sqrt(3))+t*(5/4*sqrt(3)), (1-t)*
              (-5/4)+t*(5/4),3.5],t=0..1,v=0..1,color=tan):

```

```

> f12:=plot3d([(1-t)*(5/4*sqrt(3))+t*0,(1-t)*(5/4)+t*5/2,3.5],
              t=0..1,v=0..1,color=tan):
> a13:=spacecurve([(6/4*A)*cos(t)+0,(6/4*A)*sin(t)+0,4,
                  t=0..2*Pi],color=aquamarine):
> a14:=plot3d([(1-t)*0+t*(7/4*A),(1-t)*(7/4*A)+t*0,4.5],
              t=0..1,v=0..1,color=khaki):
> b14:=plot3d([(1-t)*0+t*(-7/4*A),(1-t)*(7/4*A)+t*0,4.5],
              t=0..1,v=0..1,color=khaki):
> c14:=plot3d([(1-t)*0+t*(-7/4*A),(1-t)*(-7/4*A)+t*0,4.5],
              t=0..1,v=0..1,color=khaki):
> d14:=plot3d([(1-t)*0+t*(7/4*A),(1-t)*(-7/4*A)+t*0,4.5],
              t=0..1,v=0..1,color=khaki):
> a15:=spacecurve([(6/4*A)*cos(t)+0,(6/4*A)*sin(t)+0,5,
                  t=0..2*Pi],color=aquamarine):
> a16:=plot3d([(1-t)*0+t*0,(1-t)*(-A)+t*(-5/4*A),(1-t)*0+
              t*(0.5)],t=0..1,v=0..1,color=tan):
> a17:=plot3d([(1-t)*0+t*0,(1-t)*(A)+t*(5/4*A),(1-t)*0+
              t*(0.5)],t=0..1,v=0..1,color=tan):
> a18:=plot3d([(1-t)*(-sqrt(3))+t*(-A),(1-t)*1+t*(5/4),
              (1-t)*0+t*(0.5)],t=0..1,v=0..1,color=tan):
> a19:=plot3d([(1-t)*(-sqrt(3))+t*(-A),(1-t)*(-1)+t*(-5/4),
              (1-t)*0+t*(0.5)],t=0..1,v=0..1,color=tan):
> a20:=plot3d([(1-t)*sqrt(3)+t*A,(1-t)*(1)+t*(5/4),(1-t)*0+
              t*(0.5)],t=0..1,v=0..1,color=tan):
> a21:=plot3d([(1-t)*sqrt(3)+t*A,(1-t)*(-1)+t*(-5/4),(1-t)*0+
              t*(0.5)],t=0..1,v=0..1,color=tan):
> b15:=spacecurve([0+0*1/2*cos(t)+0*1/2*sin(t),(-5/2)+(-1)*1/2
                  *cos(t)+0*1/2*sin(t),1+0*1/2*cos(t)+1*1/2*sin
                  (t), t=3/2*Pi..2*Pi],color="DeepSkyBlue"):
> b16:=spacecurve([(-5/4*sqrt(3))+(-1/2*sqrt(3))*1/2*cos(t)+
                  0*1/2*sin(t),(-5/4)+(-1/2)*1/2*cos(t)+0*1/2
                  *sin(t),1+0*1/2*cos(t)+1*1/2*sin(t),
                  t=3/2*Pi..2*Pi],color="DeepSkyBlue"):
> b17:=spacecurve([(-5/4*sqrt(3))+(-1/2*sqrt(3))*1/2*cos(t)+0*
                  1/2*sin(t),5/4+(1/2)*1/2*cos(t)+0*1/2*sin(t),
                  1+0*1/2*cos(t)+1*1/2*sin(t),t=3/2*Pi..2*Pi],c
                  olor="DeepSkyBlue"):
> b18:=spacecurve([0+0*1/2*cos(t)+0*1/2*sin(t),(5/2)+1*1/2*
                  cos(t)+0*1/2*sin(t),1+0*1/2*cos(t)+1*1/2*sin(
                  t), t=3/2*Pi..2*Pi],color="DeepSkyBlue"):
> b19:=spacecurve([(5/4*sqrt(3))+1/2*sqrt(3))*1/2*cos(t)+0*
                  1/2*sin(t),(5/4)+(1/2)*1/2*cos(t)+0*1/2*sin(t)
                  ),1+0*1/2*cos(t)+1*1/2*sin(t),t=3/2*Pi..2*Pi]
                  ,color="DeepSkyBlue"):
> b20:=spacecurve([(5/4*sqrt(3))+1/2*sqrt(3))*1/2*cos(t)+0*
                  1/2*sin(t),(-5/4)+(-1/2)*1/2*cos(t)+0*1/2*

```

```

sin(t), 1+0*1/2*cos(t)+1*1/2*sin(t),
t=3/2*Pi..2*Pi], color="DeepSkyBlue"):
> b21:=plot3d([(1-t)*0+t*0, (1-t)*(-6/4*A)+t*(-7/4*A), (1-t)*
1+t*3/2], t=0..1, v=0..1, color=cyan):
> b22:=plot3d([(1-t)*0+t*0, (1-t)*(6/4*A)+t*(7/4*A), (1-t)*
1+t*3/2], t=0..1, v=0..1, color=cyan):
> b23:=plot3d([(1-t)*((-3/2)*sqrt(3))+t*(-7/2), (1-t)*(3/4*A)+
t*0, (1-t)*1+t*3/2], t=0..1, v=0..1, color=cyan):
> b24:=plot3d([(1-t)*((-3/2)*sqrt(3))+t*(-7/2), (1-t)*(-3/4*A)+
t*0, (1-t)*1+t*3/2], t=0..1, v=0..1, color=cyan):
> b25:=plot3d([(1-t)*((3/2)*sqrt(3))+t*(7/2), (1-t)*(-3/4*A)+
t*0, (1-t)*1+t*3/2], t=0..1, v=0..1, color=cyan):
> b26:=plot3d([(1-t)*((3/2)*sqrt(3))+t*(7/2), (1-t)*(3/4*A)+
t*0, (1-t)*1+t*3/2], t=0..1, v=0..1, color=cyan):
> g1:=plot3d([(1-t)*0+t*0, (1-t)*(-7/4*A)+t*(-6/4*A), (1-t)*
3/2+t*2], t=0..1, v=0..1, color=cyan):
> g2:=plot3d([(1-t)*0+t*0, (1-t)*(7/4*A)+t*(6/4*A), (1-t)*
3/2+t*2], t=0..1, v=0..1, color=cyan):
> g3:=plot3d([(1-t)*(-7/2)+t*((-3/2)*sqrt(3)), (1-t)*0+t*
(3/4*A), (1-t)*3/2+t*2], t=0..1, v=0..1, color=cyan):
> g4:=plot3d([(1-t)*(-7/2)+t*((-3/2)*sqrt(3)), (1-t)*0+t*
(-3/4*A), (1-t)*3/2+t*2], t=0..1, v=0..1,
color=cyan):
> g5:=plot3d([(1-t)*(7/2)+t*((3/2)*sqrt(3)), (1-t)*0+t*
(-3/4*A), (1-t)*3/2+t*2], t=0..1, v=0..1,
color=cyan):
> g6:=plot3d([(1-t)*(7/2)+t*((3/2)*sqrt(3)), (1-t)*0+t*
(3/4*A), (1-t)*3/2+t*2], t=0..1, v=0..1, color=cyan):
> h1:=spacecurve([0+0*1/2*cos(t)+0*1/2*sin(t), (-5/2)+(-1)*
1/2*cos(t)+0*1/2*sin(t), 2+0*1/2*cos(t)+1*1/2*s
in(t), t=0..1/2*Pi], color="DeepSkyBlue"):
> h2:=spacecurve([(-5/4*sqrt(3))+(-1/2*sqrt(3))*1/2*cos(t)+0*
1/2*sin(t), (-5/4)+(-
1/2)*1/2*cos(t)+0*1/2*sin(t), 2+0*1/2*
cos(t)+1*1/2*sin(t), t=0..1/2*Pi], color="DeepSk
yBlue"):
> h3:=spacecurve([(-5/4*sqrt(3))+(-1/2*sqrt(3))*1/2*cos(t)+0*
1/2*sin(t), 5/4+(1/2)*1/2*cos(t)+0*1/2*sin(t), 2
+0*1/2*cos(t)+1*1/2*sin(t), t=0..1/2*Pi], color=
"DeepSkyBlue"):
> h4:=spacecurve([0+0*1/2*cos(t)+0*1/2*sin(t), (5/2)+1*1/2*
cos(t)+0*1/2*sin(t), 2+0*1/2*cos(t)+1*1/2*sin(t)
), t=0..1/2*Pi], color="DeepSkyBlue"):
> h5:=spacecurve([(5/4*sqrt(3))+1/2*sqrt(3))*1/2*cos(t)+0*
1/2*sin(t), (5/4)+(1/2)*1/2*cos(t)+0*1/2*sin(t)
, 2+0*1/2*cos(t)+1*1/2*sin(t), t=0..1/2*Pi], colo
r="DeepSkyBlue"):

```

```

> h6:=spacecurve([(5/4*sqrt(3))+(1/2*sqrt(3))*1/2*cos(t)+0*
1/2*sin(t),(-5/4)+(-
1/2)*1/2*cos(t)+0*1/2*sin(t),2+0*1/2*
cos(t)+1*1/2*sin(t),t=0..1/2*Pi],color="DeepSk
yBlue"):
> c16:=plot3d([(1-t)*0+t*0,(1-t)*(-5/4*A)+t*(-A),(1-t)*2.5+
t*3],t=0..1,v=0..1,color=tan):
> c17:=plot3d([(1-t)*0+t*0,(1-t)*(5/4*A)+t*A,(1-t)*2.5+t*3],
t=0..1,v=0..1,color=tan):
> c18:=plot3d([(1-t)*(-A)+t*(-sqrt(3)),(1-t)*(5/4)+t*1,(1-t)*
2.5+t*3],t=0..1,v=0..1,color=tan):
> c19:=plot3d([(1-t)*(-A)+t*(-sqrt(3)),(1-t)*(-5/4)+t*(-1),
(1-t)*2.5+t*3],t=0..1,v=0..1,color=tan):
> c20:=plot3d([(1-t)*A+t*sqrt(3),(1-t)*(5/4)+t*1,(1-t)*2.5+
t*3],t=0..1,v=0..1,color=tan):
> c21:=plot3d([(1-t)*A+t*sqrt(3),(1-t)*(-5/4)+t*(-1),(1-t)*
2.5+t*3],t=0..1,v=0..1,color=tan):
> d16:=plot3d([(1-t)*0+t*0,(1-t)*(-A)+t*(-5/4*A),(1-t)*3+
t*(3.5)],t=0..1,v=0..1,color=tan):
> d17:=plot3d([(1-t)*0+t*0,(1-t)*(A)+t*(5/4*A),(1-t)*3+
t*(3.5)],t=0..1,v=0..1,color=tan):
> d18:=plot3d([(1-t)*(-sqrt(3))+t*(-A),(1-t)*1+t*(5/4),
(1-t)*3+t*(3.5)],t=0..1,v=0..1,color=tan):
> d19:=plot3d([(1-t)*(-sqrt(3))+t*(-A),(1-t)*(-1)+t*(-5/4),
(1-t)*3+t*(3.5)],t=0..1,v=0..1,color=tan):
> d20:=plot3d([(1-t)*sqrt(3)+t*A,(1-t)*1+t*(5/4),(1-t)*3+
t*(3.5)],t=0..1,v=0..1,color=tan):
> d21:=plot3d([(1-t)*sqrt(3)+t*A,(1-t)*(-1)+t*(-5/4),(1-t)*3+
t*(3.5)],t=0..1,v=0..1,color=tan):
> e15:=spacecurve([0+0*1/2*cos(t)+0*1/2*sin(t),(-5/2)+(-1)*
1/2*cos(t)+0*1/2*sin(t),4+0*1/2*cos(t)+1*1/2*
sin(t),t=3/2*Pi..2*Pi],color="DeepSkyBlue"):
> e16:=spacecurve([(-5/4*sqrt(3))+(-1/2*sqrt(3))*1/2*cos(t)+
0*1/2*sin(t),(-5/4)+(-1/2)*1/2*cos(t)+0*1/2*
sin(t),4+0*1/2*cos(t)+1*1/2*sin(t),
t=3/2*Pi..2*Pi],color="DeepSkyBlue"):
> e17:=spacecurve([(-5/4*sqrt(3))+(-1/2*sqrt(3))*1/2*cos(t)+
0*1/2*sin(t),5/4+(1/2)*1/2*cos(t)+0*1/2*sin(t)
),4+0*1/2*cos(t)+1*1/2*sin(t),t=3/2*Pi..2*Pi]
,color="DeepSkyBlue"):
> e18:=spacecurve([0+0*1/2*cos(t)+0*1/2*sin(t),(5/2)+1*1/2*
cos(t)+0*1/2*sin(t),4+0*1/2*cos(t)+1*1/2*
sin(t),t=3/2*Pi..2*Pi],color="DeepSkyBlue"):
> e19:=spacecurve([(5/4*sqrt(3))+(1/2*sqrt(3))*1/2*cos(t)+0*
1/2*sin(t),(5/4)+(1/2)*1/2*cos(t)+0*1/2*
sin(t),4+0*1/2*cos(t)+1*1/2*sin(t),
t=3/2*Pi..2*Pi],color="DeepSkyBlue"):

```

```

> e20:=spacecurve([(5/4*sqrt(3))+(1/2*sqrt(3))*1/2*cos(t)+0*
1/2*sin(t), (-5/4)+(-1/2)*1/2*cos(t)+0*
1/2*sin(t), 4+0*1/2*cos(t)+1*1/2*sin(t),
t=3/2*Pi..2*Pi], color="DeepSkyBlue"):
> e21:=plot3d([(1-t)*0+t*0, (1-t)*(-6/4*A)+t*(-7/4*A), (1-t)*4+
t*9/2], t=0..1, v=0..1, color=cyan):
> e22:=plot3d([(1-t)*0+t*0, (1-t)*(6/4*A)+t*(7/4*A), (1-t)*4+
t*9/2], t=0..1, v=0..1, color=cyan):
> e23:=plot3d([(1-t)*((-3/2)*sqrt(3))+t*(-7/2), (1-t)*(3/4*A)+
t*0, (1-t)*4+t*9/2], t=0..1, v=0..1, color=cyan):
> e24:=plot3d([(1-t)*((-3/2)*sqrt(3))+t*(-7/2), (1-t)*(-3/4*A)+
t*0, (1-t)*4+t*9/2], t=0..1, v=0..1, color=cyan):
> e25:=plot3d([(1-t)*((3/2)*sqrt(3))+t*(7/2), (1-t)*(-3/4*A)+
t*0, (1-t)*4+t*9/2], t=0..1, v=0..1, color=cyan):
> e26:=plot3d([(1-t)*((3/2)*sqrt(3))+t*(7/2), (1-t)*(3/4*A)+
t*0, (1-t)*4+t*9/2], t=0..1, v=0..1, color=cyan):
> g7:=plot3d([(1-t)*0+t*0, (1-t)*(-7/4*A)+t*(-6/4*A), (1-t)*9/2+
t*5], t=0..1, v=0..1, color=cyan):
> g8:=plot3d([(1-t)*0+t*0, (1-t)*(7/4*A)+t*(6/4*A), (1-t)*9/2+
t*5], t=0..1, v=0..1, color=cyan):
> g9:=plot3d([(1-t)*(-7/2)+t*((-3/2)*sqrt(3)), (1-t)*0+t*
(3/4*A), (1-t)*9/2+t*5], t=0..1, v=0..1, color=cyan):
> g10:=plot3d([(1-t)*(-7/2)+t*((-3/2)*sqrt(3)), (1-t)*0+t*
(-3/4*A), (1-t)*9/2+t*5], t=0..1, v=0..1,
color=cyan):
> g11:=plot3d([(1-t)*(7/2)+t*((3/2)*sqrt(3)), (1-t)*0+t*
(-3/4*A), (1-t)*9/2+t*5], t=0..1, v=0..1,
color=cyan):
> g12:=plot3d([(1-t)*(7/2)+t*((3/2)*sqrt(3)), (1-t)*0+t*
(3/4*A), (1-t)*9/2+t*5], t=0..1, v=0..1, color=cyan):

```

B.3 Kerangka Lontin Cekung dan Interpolasi

```

> i1:=plot3d([(1-v)*((1-t)*0+t*(-sqrt(3)))+v*((1-t)*0+t*(3/4*
(-sqrt(3))))], (1-v)*((1-t)*A+t*1)+v*((1-t)*(3/4*A)+
t*3/4), (1-v)*((1-t)*0+t*0)+v*0.5], t=0..1, v=0..1,
color="DeepSkyBlue"):
> i2:=plot3d([(1-v)*((1-t)*(-sqrt(3))+t*0)+v*((1-t)*(3/4*
(-sqrt(3)))+t*0), (1-v)*((1-t)*(-1)+t*(-2))+v*
((1-t)*(-3/4)+t*(-6/4)), (1-v)*((1-
t)*0+t*0)+v*0.5], t=0..1, v=0..1,
color="DeepSkyBlue"):
> i3:=plot3d([(1-v)*((1-t)*sqrt(3)+t*sqrt(3))+v*((1-t)*
(3/4*sqrt(3))+t*(3/4*sqrt(3))), (1-v)*((1-t)*
(-1)+t*1)+v*((1-t)*(-3/4)+t*3/4), (1-v)*
((1-t)*0+t*0)+v*0.5], t=0..1, v=0..1,
color="DeepSkyBlue"):

```

```

> i4:=plot3d([(1-v)*(-0.5)+v*((1/2*A)*cos(t*(2/6*Pi)+(5/6*Pi))
+0),(1-v)*(0)+v*((1/2*A)*sin(t*(2/6*Pi)+(5/6*Pi))
+0),(1-v)*3/2+v*1],t=0..1,v=0..1,color=pink):
> i5:=plot3d([(1-v)*(0.5)+v*((1/2*A)*cos(t*(2/6*Pi)+(11/6*Pi))
+0),(1-v)*(0)+v*((1/2*A)*sin(t*(2/6*Pi)+(11/6*Pi))
+0),(1-v)*3/2+v*1],t=0..1,v=0..1,
color=pink):
> i6:=plot3d([(1-v)*(-0.5)+v*((1/2*A)*cos(t*(2/6*Pi)+(5/6*Pi))
+0),(1-v)*(0)+v*((1/2*A)*sin(t*(2/6*Pi)+(5/6*Pi))
+0),(1-v)*3/2+v*2],t=0..1,v=0..1,color=pink):
> i7:=plot3d([(1-v)*(0.5)+v*((1/2*A)*cos(t*(2/6*Pi)+(11/6*Pi))
+0),(1-v)*(0)+v*((1/2*A)*sin(t*(2/6*Pi)+(11/6*Pi))
+0),(1-v)*3/2+v*2],t=0..1,v=0..1,color=pink):
> i8:=plot3d([(1-v)*((1-t)*0+t*(3/4*(-sqrt(3))))+v*((1-t)*0+t*
(-sqrt(3))), (1-v)*((1-t)*(3/4*A)+t*3/4)+v*((1-
t)*A+t*1), (1-v)*(5/2)+v*3],t=0..1,v=0..1,
color="DeepSkyBlue"):
> i9:=plot3d([(1-v)*((1-t)*(3/4*(-sqrt(3)))+t*0)+v*((1-t)*
(-sqrt(3))+t*0), (1-v)*((1-t)*(-3/4)+t*(-6/4))
+v*((1-t)*(-1)+t*(-2)), (1-v)*(5/2)+v*3],
t=0..1,v=0..1,color="DeepSkyBlue"):
> i10:=plot3d([(1-v)*((1-t)*(3/4*sqrt(3))+t*(3/4*sqrt(3)))+v*
((1-t)*sqrt(3)+t*sqrt(3)), (1-v)*((1-t)*(-3/4)+t*
3/4)+v*((1-t)*(-1)+t*1), (1-v)*(5/2)+v*3],
t=0..1,v=0..1,color="DeepSkyBlue"):
> i11:=plot3d([(1-v)*((1-t)*0+t*(-sqrt(3)))+v*((1-t)*0+t*(3/4*
(-sqrt(3))))), (1-v)*((1-t)*A+t*1)+v*((1-t)*(3/4*A)+
t*3/4), (1-v)*3+v*7/2],t=0..1,v=0..1,color=cyan):
> i12:=plot3d([(1-v)*((1-t)*(-sqrt(3))+t*0)+v*((1-t)*(3/4*
(-sqrt(3)))+t*0), (1-v)*((1-t)*(-1)+t*(-2))+v*
((1-t)*(-3/4)+t*(-6/4)), (1-v)*3+v*7/2],
t=0..1,v=0..1,color=cyan):
> i13:=plot3d([(1-v)*((1-t)*sqrt(3)+t*sqrt(3))+v*((1-t)*
(3/4*sqrt(3))+t*(3/4*sqrt(3))), (1-v)*((1-t)*
(-1)+t*1)+v*((1-t)*(-3/4)+t*3/4), (1-v)*3+v*7/2],
t=0..1,v=0..1,color=cyan):

```