# MODELISASI TIANG TERAS MENGGUNAKAN HASIL DEFORMASI PRISMA SEGIENAM, TABUNG, DAN BOLA

**SKRIPSI**

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

Oleh

**Elsha Henik Sugianto**
**NIM 141810101051**

**JURUSAN MATEMATIKA**
**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**
**UNIVERSITAS JEMBER**
**2018**

**MODELISASI TIANG TERAS MENGGUNAKAN HASIL DEFORMASI PRISMA SEGIENAM, TABUNG, DAN BOLA**

**SKRIPSI**

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

Oleh

**Elsha Henik Sugianto**
**NIM 141810101051**

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

**2018**

# PERSEMBAHAN

Puji Tuhan, dengan hormat syukurku kunaikan untuk Tuhan Yesus Kristus, skripsi ini saya persembahkan untuk:

1. Papa Agun dan Mama Marta tercinta, yang telah mendoakan dan memberi kasih sayang serta pengorbanan untuk putri tercintanya;
2. kakak dan adik tersayang Onne Hena S. S.Si., dan Angelina HS. yang telah banyak memberikan semangat dalam penyelesaian skripsi ini;
3. teman-teman yang telah memberikan dukungan dalam penyelesaian skripsi ini;
4. guru-guruku sejak taman kanak-kanak sampai perguruan tinggi, yang telah memberikan ilmu dan membimbing dengan penuh kesabaran;
5. Almamater Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember, SMA Kemala Bhayangkari 3 Porong, SMP Negeri 1 Porong, SD Kemala Bhayangkari Porong, dan TK Kemala Bhayangkari Porong.

**MOTTO**

Mintalah, maka akan diberikan kepadamu; carilah, maka kamu akan mendapat; ketoklah, maka pintu akan dibukakan bagimu. Karena setiap orang yang meminta, menerima dan setiap orang yang mencari, mendapat dan setiap orang yang mengetok, baginya pintu akan dibukakan
(Matius 7:7-8)[1]

Segala perkara dapat kutanggung di dalam Dia yang memberi kekuatan kepadaku
(Filipi 3:14)[1]

---

[1] *Alkitab*. Lembaga Alkitab Indonesia. Jakarta: Lembaga Alkitab Indonesia.

# PERNYATAAN

Saya yang bertanda tangan di bawah ini :

Nama : Elsha Henik Sugianto

NIM : 141810101051

Menyatakan dengan sesungguhnya bahwa skripsi yang berjudul “Modelisasi Tiang Teras Menggunakan Hasil Deformasi Prisma Segienam, Tabung dan Bola” adalah benar-benar hasil karya sendiri, kecuali kutipan yang sudah saya sebutkan sumbernya, belum pernah diajukan pada institusi manapun, dan bukan karya jiplakan. Saya bertanggung jawab atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

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

Jember, Oktober 2018

Yang menyatakan,

Elsha Henik Sugianto

NIM 141810101051

**SKRIPSI**

**MODELISASI TIANG TERAS MENGGUNAKAN HASIL DEFORMASI PRISMA SEGIENAM, TABUNG, DAN BOLA**

Oleh
Elsha Henik Sugianto
NIM. 141810101051

Pembimbing

Dosen Pembimbing Utama : Bagus Juliyanto, S.Si, M.Si.
Dosen Pembimbing Anggota : Dr. Firdaus Ubaidillah, S.Si., M.Si.

# PENGESAHAN

Skripsi berjudul "Modelisasi Tiang Teras Menggunakan Hasil Deformasi Prisma Segienam, Tabung dan Bola" telah diuji dan disahkan pada:

hari : 
tanggal : 
tempat : Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember

Tim Penguji :

| Ketua, | Anggota I, |
|---|---|
| Bagus Juliyanto, S.Si, M.Si.<br>NIP 19800702 200312 1 001 | Dr. Firdaus Ubaidillah, S.Si., M.Si.<br>NIP 19700606 199803 1 003 |
| Anggota II, | Anggota III, |
| Ika Hesti Agustin, S.Si., M.Si.<br>NIP 19840801 200801 2 006 | Ikhsanul Halikin, S.Pd., M.Si.<br>NIP 19861014 201404 1 001 |

Mengesahkan

Dekan,

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

# RINGKASAN

**Modelisasi Tiang Teras Menggunakan Hasil Deformasi Prisma Segienam, Tabung dan Bola**; Elsha Henik Sugianto ; 141810101051; 2018; 65 Halaman; Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

Pada konstruksi sebuah bangunan, tiang atau pilar berfungsi sebagai penyangga dan penguat suatu bangunan. Tidak hanya itu, tiang dapat difungsikan sebagai obyek yang dapat menambah nilai keindahan suatu bangunan. Berdasarkan pengamatan dilapangan, model tiang teras pada umumnya masih memiliki kekurangan pada tampilan bentuk, contohnya terdiri dari satu benda ruang seperti tabung, kubus, atau balok saja sehingga terkesan kurang variatif. Bentuk-bentuk geometris yang variatif dapat diperoleh dengan menggunakan beberapa teknik diantaranya menggunakan teknik memutar kurva, ataupun deformasi. Tujuan dari penelitian ini adalah untuk mendapatkan beragam bentuk desain tiang teras yang variatif dari penggabungan hasil deformasi benda-benda geometri ruang.

Modelisasi tiang teras dibagi menjadi tiga tahapan seperti berikut. Pertama, membangun benda dasar sebagai komponen penyusun tiang teras dari hasil deformasi prisma segienam, tabung, dan bola. Dalam hal ini benda geometri tersebut dioperasikan melalui titik dan kurva kemudian membangun permukaan lengkung atau menginterpolasikan kurva tersebut. Kedua, merangkai beberapa benda dasar komponen tiang teras pada sumbu pemodelan. Tahapan terakhir dilakukan programasi untuk memodelisasi tiang teras tersebut dengan bantuan *software* Maple 18.

Dari hasil penelitian ini didapatkan dua prosedur untuk memodelisasi tiang teras. Pertama, prosedur untuk mendesain beragam bentuk komponen penyusun tiang teras dari benda dasar prisma segienam, tabung, dan bola adalah dilakukan dengan cara menetapkan bidang yang akan dijadikan bidang potong untuk prisma segienam, tabung, dan bola kemudian menggunakan bidang-bidang tersebut untuk

memotong benda-benda geometri sehingga menghasilkan benda-benda geometri terpancung. Setelah itu menetapkan titik-titik atau pola pada tiap sisi tegak atau tutup prisma, tabung dan bola. Kemudian mengoperasikan titik-titik tersebut dengan ; (a) menetapkan titik kontrol untuk memperkecil atau memperbesar jari-jari dan ketinggian; (b) membangun segmen garis, bidang lingkaran, kurva bezier kuadratik; (c) menginterpolasikan kurva atau memutar kurva sehingga menghasilkan bentuk komponen tiang teras.

Kedua, prosedur untuk merangkai komponen penyusun tiang teras hasil dari prosedur pertama pada sumbu pemodelan dengan cara membagi sumbu menjadi tiga segmen yang diperlukan sebagai sumbu bagian alas, bagian tengah, dan bagian atas. Kemudian mengisi setiap bagian segmen sumbu non homogen tersebut dengan komponen penyusun tiang teras sehingga menghasilkan model tiang teras yang bervariasi.

# PRAKATA

Puji Tuhan kunaikan syukur pada Tuhan Yesus, yang telah memberikan Kasih-Nya yang tidak pernah berkesudahan, yang selalu setia menuntun dan membimbing setiap waktu sehingga penulis dapat menyelesaikan skripsi yang berjudul ”Modelisasi Tiang Teras Dengan Hasil Deformasi Prisma Segienam, Tabung, dan Bola”. 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. Bagus Juliyanto, S.Si, M.Si. selaku Dosen Pembimbing Utama dan Dr. Firdaus Ubaidillah, S.Si, M.Si selaku Dosen Pembimbing Anggota yang telah meluangkan waktu, pikiran, dan perhatian dalam penulisan skripsi ini;
2. Ika Hesti Agustin, S.Si, M.Si dan Ikhsanul Halikin, S.Si.,M.Si selaku Dosen Penguji yang telah memberikan kritikan dan saran demi kesempurnaan skripsi ini;
3. Papa Agun dan Mama Marta tercinta, yang banyak mendoakan, memberi motivasi dan tidak lupa memberi kasih sayang serta pengorbanan untuk putri tercintanya;
4. kakak tersayang Onne Hena S., S.Si. dan adik tersayang Angelina HS. yang telah banyak memberikan semangat dan pemikirannya dalam membantu menyelesaian skripsi ini;
5. sahabat-sahabatku Ulfi, Puni, Lia, Frisca, Dinda, Nisa, Anin, dan Rofi, terima kasih atas kebersamaan selama waktu kuliah, dan tidak lupa memberikan semangat dan motivasi dalam menyelesaikan skripsi ini;
6. teman-teman seperjuangan Geometri Rancang Bangun (Nurika, Dita, Lisma, Edo, Fay, dan Dinar) yang telah membantu dan memberikan semangat selama mengerjakan skripsi ini
7. 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, Oktober 2018 Penulis

# DAFTAR ISI

# DAFTAR GAMBAR

Halaman

# DAFTAR TABEL

# DAFTAR LAMPIRAN

Halaman

# BAB 1. PENDAHULUAN

## 1.1 Latar Belakang

Tiang menurut Kamus Besar Bahasa Indonesia adalah tonggak panjang yang terbuat dari besi, kayu, dan sebagainya yang dipancangkan untuk suatu keperluan misalkan pada rumah yang disebut tiang rumah. Pada konstruksi sebuah bangunan, tiang atau pilar berfungsi sebagai penyanggah dan penguat suatu bangunan. Tidak hanya itu, tiang dapat difungsikan sebagai obyek yang dapat menambah nilai keindahan suatu bangunan. Pada bagian rumah, tiang terpasang di bagian depan atau bagian teras rumah. Tiang teras  pada umumnya terbuat dari komponen kerangka tertutup kemudian diisi dengan berbagai bahan bangunan seperti campuran semen, pasir, batu dan bahan-bahan lainnya. Secara umum tiang teras terdiri dari tiga bagian yaitu bagian alas, tengah dan bagian atas (Gambar 1.1). Bagian alas  dan bagian atas tiang teras biasanya tersusun dari komponen benda geometri yang bentuk permukaannya lebih luas atau lebih besar karena dibutuhkan sebagai pondasi yang kuat untuk menyanggah bagian tengah agar tetap seimbang. Dari model tiang teras yang sudah ada (Gambar 1.1), bentuk tiang teras masih memiliki model yang sederhana dan menggunakan hanya satu bentuk benda geometri ruang pada bagian alas, tengah, dan atas sehingga tampilan bentuk tiang teras kurang menarik dan kurang bervariatif.


Sumber : http://rumahminimalisoi.com/

Gambar 1.1 Contoh bentuk desain tiang teras

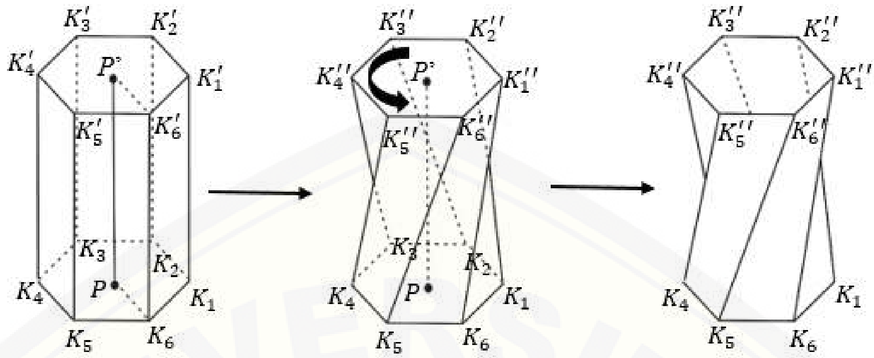
Beberapa penelitian sebelumnya Santoso (2003) telah melakukan penelitian tentang Modelisasi Bentuk Blok Aksesoris Bangunan Berupa Tiang Teras, Roster, dan Pot Bunga yang membahas masalah prosedur konstruksi tiang teras dari balok, tabung, bola dan gabungannya. Namun pada penelitian tersebut tiang teras yang terbentuk hanya terdiri dari satu bentuk bangun ruang yang ditata berupa tingkatan dengan menggunakan transformasi dan tidak terdapat modifikasi bentuk pada tiap bangun ruang yang digunakan. Kemudian Fatkurotin (2015) meneliti tentang Konstruksi Botol Parfum melalui Penggabungan Hasil Deformasi Prisma, Bola, dan Tabung. Pada penelitian botol parfum yang menggunakan benda dasar geometri prisma segiempat, tabung dan bola, hasil deformasi yang diperoleh dari hasil penggabungan pemodelan cukup aplikatif digunakan untuk memodelisasi bagian-bagian tiang teras akan tetapi hasil deformasi pada tiap benda memiliki kekurangan misalnya pemotongan pada benda geometri hanya dilakukan satu kali saja sehingga variasi bentuk yang didapat kurang beragam dan kurang memberikan variasi pada tiap bangun ruang yang digunakan. Pada konteks geometri deformasi adalah mengubah bentuk atau tampilan benda awal menjadi benda yang baru dengan memberikan perlakuan atau metode pada benda awal.

Berdasarkan hal tersebut, penelitian ini dimaksudkan untuk memodelisasi tiang teras dengan mengunakan deformasi dengan teknik penggabungan, teknik pemotongan dan interpolasi pada benda geometri berupa prisma segienam beraturan, tabung dan bola. Pemilihan bangun ruang berupa prisma segienam, tabung, dan bola dimaksudkan untuk memperhatikan kesimetrisan bangun akan dihasilkan, selain itu pemilihan bola jarang digunakan sebelumnya pada penelitian sebelumnya.

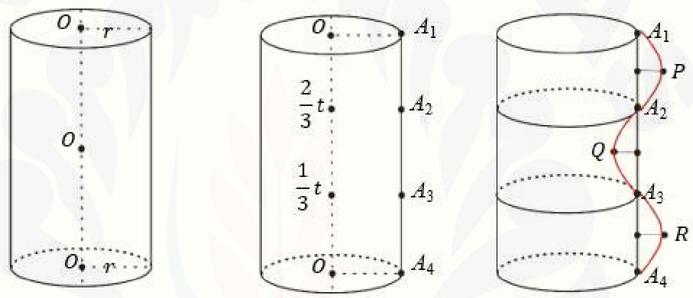
### 1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dipaparkan sebelumnya, adapun rumusan masalah pada permasalahan modelisasi tiang teras rumah ini adalah sebagai berikut

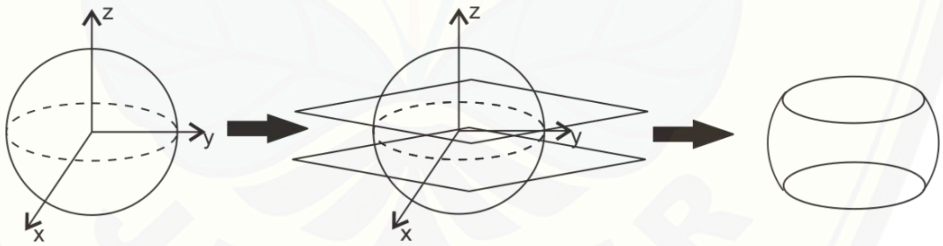
a. Diberikan prisma segienam beraturan, tabung dan bola. Dari ketiga benda geometri ruang tersebut, bagaimana prosedur untuk membangun beberapa benda dasar sebagai komponen penyusun tiang teras dari hasil deformasi prisma segienam beraturan, tabung dan bola sehingga menghasilkan komponen penyusun bagian tiang teras yang bervariasi (Gambar 1.2).




(a) Deformasi Prisma Segienam beraturan



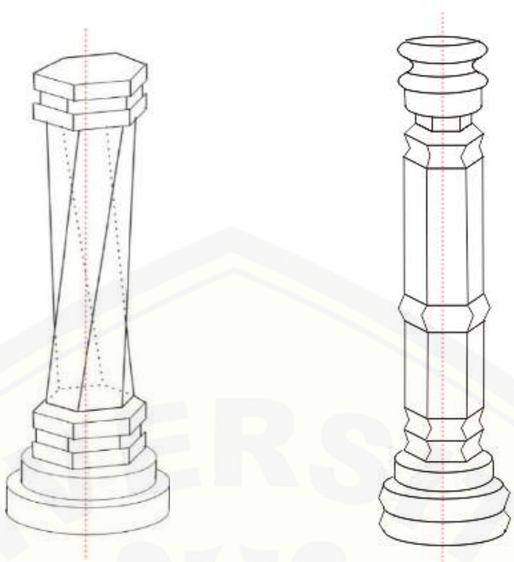

(b) Deformasi Tabung




(c) Deformasi Bola

Gambar 1.2 Benda dasar komponen penyusun tiang teras

b. Diberikan satu model kerangka pemodelan untuk merangkai tiang teras, bagaimana prosedur merangkai beberapa benda dasar komponen tiang teras agar menghasilkan model tiang teras yang tergabung secara utuh dan bervariasi (Gambar 1.3)



Gambar 1.3 Contoh hasil penggabungan membentuk variasi tiang teras

**1.3 Tujuan**

Tujuan dari penelitian pemodelan tiang teras ini adalah sebagai berikut

a. Mendapatkan prosedur untuk membangun beberapa benda dasar sebagai komponen penyusun konstruksi bentuk tiang teras dari pemotongan pada bidang datar dan lengkung pada prisma segienam beraturan, tabung, dan bola.
b. Mendapat prosedur untuk merangkai benda dasar sebagai penyusun bentuk tiang teras pada satu sumbu penggabungan dan menyusunnya menjadi satu bentuk tiang teras yang utuh.

**1.4 Manfaat**

Adapun manfaat yang diperoleh dalam pemodelan tiang teras ini adalah sebagai berikut

a. Menggunakan bantuan komputer sehingga dapat dihasilkan beberapa model bentuk tiang teras yang bervariasi, simetris dan indah.
b. Memberikan informasi kepada pembaca tentang beberapa model bentuk tiang teras sehingga dapat menambah pilihan bentuk tiang teras yang sudah ada sebelumnya.

# BAB 2. TINJAUAN PUSTAKA

Dari beberapa persoalan yang dimaksud pada Bab 1 dan untuk mencari solusi dari permasalahan modelisasi bentuk tiang teras, maka pada bab ini akan disajikan beberapa teori dasar yang berkaitan dengan prosedur modelisasi tiang teras. Teori dasar mengenai pemodelisasian tiang teras tersebut meliputi kajian tentang segmen garis, lingkaran, transformasi di $R^3$, dan kurva Bezier serta benda-benda ruang geometri seperti prisma segienam, tabung dan bola. Hal ini bertujuan untuk mempermudah dalam proses modelisasi beragam komponen bagian tiang teras dan perangkaian bagian-bagian tiang teras.

## 2.1 Penyajian Segmen Garis, Lingkaran, dan Poligon Segienam

### 2.1.1 Penyajian Segmen Garis

Menurut Kusno (2003) $\overline{AB}$ merupakan penyajian segmen garis AB yang merupakan himpunan titik-titik dari garis yang memuat titik *A*, titik *B*, dan semua titik diantara titik *A* dan titik *B*. Misal diberikan dua buah titik berbeda di ruang dengan koordinat masing-masing $A(x_1, y_1, z_1)$ dan $B(x_2, y_2, z_2)$ maka segmen garis $\overline{AB}$ didefinisikan sebagai tempat kedudukan titik-titik $C(x, y, z)$ seperti pada Gambar 2.1. Persamaan vektorial segmen garis $\overline{AB}$ dapat didefinisikan sebagai berikut

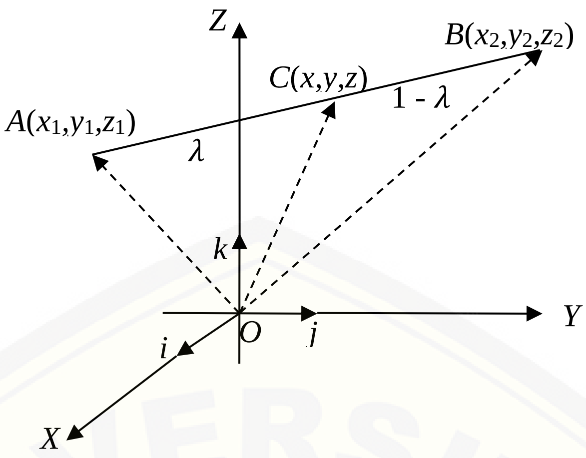
$$\overrightarrow{OC} = \lambda\overrightarrow{OB} + (1\text{-}\lambda)\overrightarrow{OA} \qquad (2.1a)$$

dengan $\lambda \in [0,1]$ sebagai variabel parameter dan $C \in \overline{AB}$. Dengan demikian diperoleh persamaan parametrik segmen garis dapat dinyatakan sebagai:

$$\langle x,y,z\rangle = \lambda\langle x_2,y_2,z_2\rangle + (1\text{-}\lambda)\langle x_1,y_1,z_1\rangle, \qquad (2.1b)$$

atau

$$\begin{aligned} x &= (1-\lambda)x_1 + \lambda x_2, \\ y &= (1-\lambda)y_1 + \lambda y_2, \\ z &= (1-\lambda)z_1 + \lambda z_2. \end{aligned} \qquad (2.1c)$$




Gambar 2.1 Penyajian segmen garis di ruang

2.1.2 Penyajian Lingkaran dan Bagiannya

Definisi lingkaran yaitu himpunan titik-titik di bidang yang berjarak sama dari titik tertentu yang disebut pusat lingkaran (Kusno, 2002). Misalkan diketahui sebarang titik $A(x,y)$ pada lingkaran yang berpusat pada $B(x_1,y_1)$, maka melalui $A$ tarik garis $g$ sejajar sumbu $Y$ dan melalui $B$ tarik garis $h$ sejajar sumbu $X$. Titik $C$ merupakan perpotongan dari kedua garis tersebut dan $\angle ACB$ membentuk sudut siku-siku (Gambar 2.2). Maka didapat hubungan:

$$\overrightarrow{BA} = \overrightarrow{BC} + \overrightarrow{CA} \tag{2.2}$$

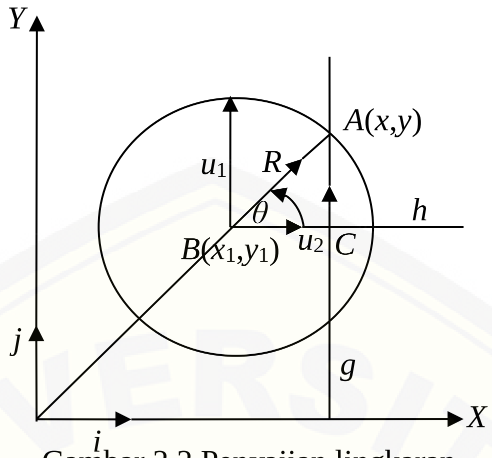
Dari Persamaan (2.2) dapat dibentuk persamaan parametrik lingkaran dengan arah vektor satuan $\boldsymbol{u_1}$ dan $\boldsymbol{u_2}$ sebagai berikut:

$$\overrightarrow{OA} - \overrightarrow{OB} = R\cos\theta\,\boldsymbol{u_1} + R\sin\theta\,\boldsymbol{u_2},$$
$$\langle x - x_1, y - y_1\rangle = \langle R\cos\theta, R\sin\theta\rangle,$$
$$\langle x,y\rangle = \langle x_1 + R\cos\theta,\ y_1 + R\sin\theta\rangle,$$

atau dapat juga ditulis (Persamaan 2.3) :

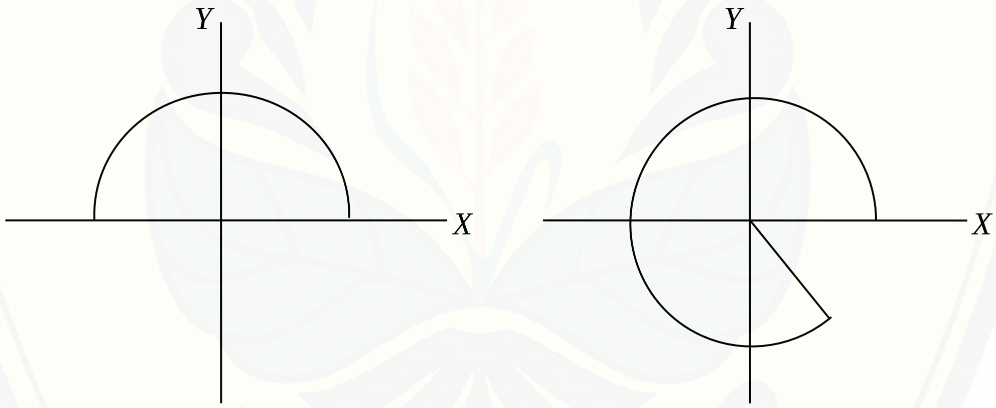
$$\begin{aligned} x(\theta) &= x_1 + R\cos\theta, \\ y(\theta) &= y_1 + R\sin\theta, \end{aligned} \tag{2.3}$$

dengan $0 \le \theta \le 2\pi$, dan $R$ merupakan jari-jari lingkaran berharga real.




Gambar 2.2 Penyajian lingkaran

Apabila parameter $\theta$ pada persamaan (2.3) diberikan nilai pada interval $\theta_1 \leq \theta \leq \theta_2$, maka akan diperoleh sebuah keratan lingkaran (Gambar 2.3).



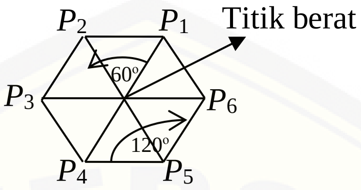

(a) Untuk $0 \leq \theta \leq \pi$ (b) Untuk $0 \leq \theta \leq \frac{10\pi}{6}$

Gambar 2.3 Penyajian keratan lingkaran

2.1.3 Penyajian Poligon Segi Enam Beraturan

Definisi 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 sehingga bila terdapat dua ruas garis sebarang yang berpotongan maka akan membentuk titik potong di salah satu titik-titik $P_1, P_2, \dots, P_{n-1}, P_n$ dan tidak ada titik lain. Poligon konveks adalah poligon yang masing-masing sudutnya lebih kecil dari 180$^{o}$ (Kusno, 2002).

Poligon segienam beraturan adalah suatu poligon konveks bersisi enam sisi dengan panjang sisi dan besar sudut yang sama. Besar sudut pada poligon segienam beraturan adalah 120º dan besar sudut pusat masing-masing adalah 60º (Gambar 2.4).




Gambar 2.4 Poligon Segienam Beraturan

Berdasarkan definisi poligon segienam beraturan tersebut, misalkan diketahui titik berat $D(0,0,z_1)$ yang terletak pada bidang $z = z_1$ dan jarak titik $D(0,0,z_1)$ ke titik-titik sudut poligon adalah $l$, sehingga dapat dibangun poligon segienam beraturan dengan langkah-langkah berikut (Gambar 2.5).

a. Menetapkan titik sudut awal $P_1(0,l,z_1)$.

b. Merotasikan titik $P_1$ terhadap titik berat dengan sudut rotasi sebesar 60º menggunakan formula:

$$\begin{bmatrix} x_{i+1} \\ y_{i+1} \\ z_{i+1} \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} \tag{2.4}$$

dan diperoleh titik $P_2(x_2,y_2,z_1)$ dengan $i = 1, \dots ,6$.

c. Dengan mempertahankan besar sudut 60º dan arah rotasi, ulangi langkah (b) untuk titik-titik $P_i$ dengan $i$ = 2, 3, ..., 6 sehingga dihasilkan titik-titik $P_3(x_3,y_3,z_1)$, $P_4(x_4,y_4,z_1)$, ..., $P_6(x_6,y_6,z_1)$.

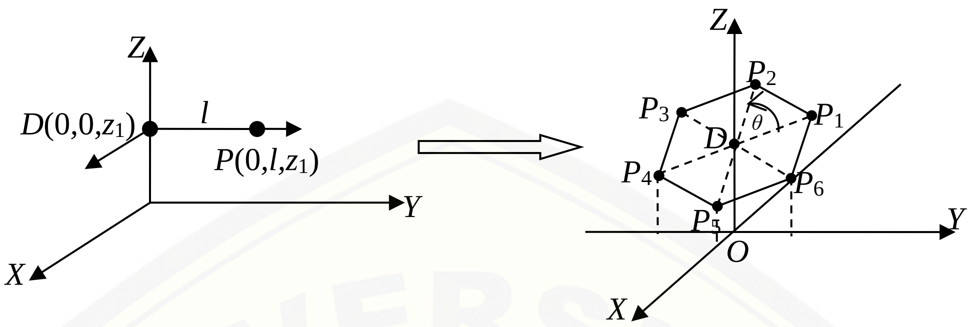
d. Membangun poligon segi enam beraturan dengan cara membuat segmen-segmen garis $\overline{P_1P_2}$, $\overline{P_2P_3}, \dots, \overline{P_5P_6}$, $\overline{P_6P_1}$, menggunakan persamaan (2.1a) (Kusno, 2002).

$P_1(x_1,y_1,z_1)$ 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 berikut

$$(1-t)\langle x_i,y_i,z_i\rangle + t\langle x_{i+1},y_{i+1},z_{i+1}\rangle = \langle x,y,z\rangle \text{ untuk } 3 \leq i < 6 \tag{2.5}$$

$$(1-t)\langle x_6,y_6,z_6\rangle + t\langle x_1,y_1,z_1\rangle = \langle x,y,z\rangle \text{ untuk } i = 6 \tag{2.6}$$

dengan $\langle x_i, y_i, z_i\rangle$ merupakan 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 2.5 Langkah-langkah Membangun Poligon Segi Enam Beraturan pada Bidang $z = z_1$

## 2.2 Interpolasi diantara Segmen Garis dan Kurva di Ruang

Misalkan terdapat dua segmen garis $\overline{AB}$ dan $\overline{CD}$ didefinisikan masing-masing oleh $A(x_1, y_1, z_1)$, $B(x_2, y_2, z_2)$, $C(x_3, y_3, z_3)$ dan $D(x_4, y_4, z_4)$ dalam bentuk parametrik $\boldsymbol{l}_1(u)$ yang merupakan bentuk parametrik segmen garis $\overline{AB}$ dan $\boldsymbol{l}_2(u)$ yang merupakan bentuk parametrik segmen garis $\overline{CD}$, maka permukaan parametrik hasil interpolasi linier kedua segmen garis tersebut diformulasikan sebagai berikut:

$$\boldsymbol{S}(u,v) = (1-v)\boldsymbol{l}_1(u) + v\boldsymbol{l}_2(u), \tag{2.7}$$

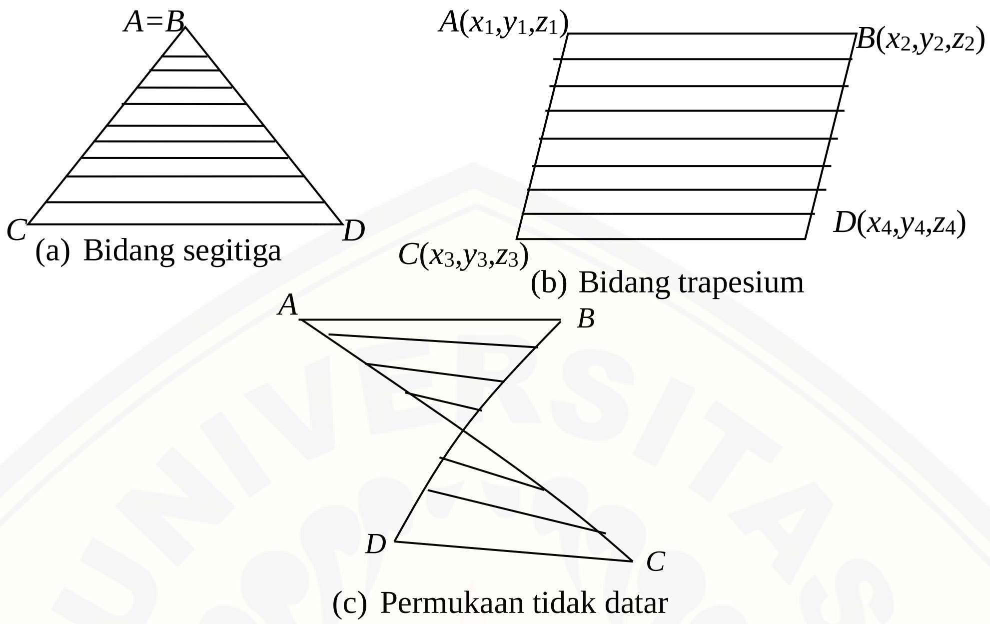
dengan $0 \leq u \leq 1$ dan $0 \leq v \leq 1$.

Menurut Astuti (2014) bahwasannya terdapat beberapa kasus khusus untuk interpolasi linier kedua garis tersebut. Misalkan jika $A$=$B$ maka hasil interpolasi Persamaan (2.7) akan menghasilkan bidang segitiga (Gambar 2.6). Sedangkan jika $\overline{AB} // \overline{CD}$ maka secara umum akan terbentuk bidang segiempat (Gambar 2.6c). Jika bidang tersebut dibentuk dari interpolasi dua garis yang bersilangan maka hasil interpolasi akan menghasilkan permukaan tidak datar (dapat melengkung ataupun terjadi puntiran di sebagian permukaan tersebut) (Gambar 2.6b).

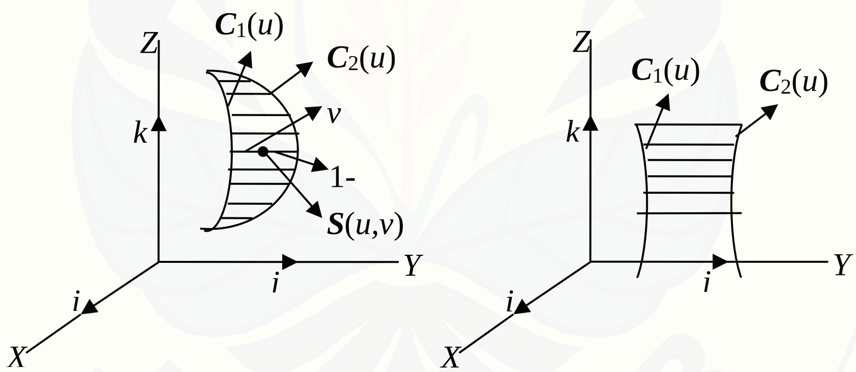
Interpolasi kurva ruang dapat membangun permukaan lengkung hasil melalui persamaan berikut:

$$\boldsymbol{S}(u,v) = (1-v)\boldsymbol{C}_1(u) + v\boldsymbol{C}_2(u), \tag{2.8}$$

dengan $\boldsymbol{C}_1(u)$ dan $\boldsymbol{C}_2(u)$ merupakan kurva batas (Gambar 2.7).




Gambar 2.6 Contoh kasus khusus interpolasi linier dua segmen garis




Gambar 2.7 Interpolasi linier pada kurva

**2.3 Penyajian Bola, Tabung, dan Prisma Segienam Beraturan**

2.3.1 Penyajian Bola

Bola adalah kedudukan titik-titik dalam ruang yang berjarak sama terhadap titik tertentu (Kusno, 2002). Titik tertentu tersebut dinamakan sebagai pusat bola, jari-jari adalah ruas garis dari pusat ke titik pada bola. Semua ruas garis penghubung dua titik pada bola yang melalui pusat disebut diameter (garis tengah). Pada bagian ini dijelaskan mengenai persamaan bola dalam bentuk parametrik.

Jika diketahui bola dengan pusat $Q(a,b,c)$ dan $PQ = r$, maka bentuk parametrik bola dapat dinyatakan sebagai berikut (Gambar 2.8).

Persamaan bola

$$B(\emptyset,\theta) = \overline{OQ} + S(\emptyset,\theta),$$

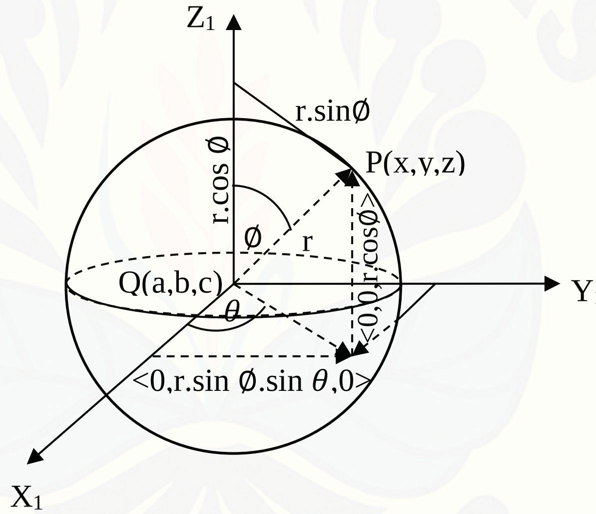
dengan $\overline{OQ} = < a,b,c >$ atau

$B(\emptyset,\theta) =< r.\sin\emptyset.\cos\theta, r.\sin\emptyset.\sin\theta, r.\cos\emptyset >.$

Dengan demikian persamaan parametrik bola dengan pusat $Q(a,b,c)$ dan jari-jari *r* adalah :

$B(\emptyset,\theta) =< r.\sin\emptyset.\cos\theta + a, r.\sin\emptyset.\sin\theta + b, r.\cos\emptyset + c >$ (2.9)

dengan $0 \le \emptyset,\theta \le 2\pi$, sedangkan *r*, *a*, *b* dan *c* adalah konstanta real.




Gambar 2.8 Bola dengan pusat $Q(a,b,c)$ dan jari-jari *r*

Pada kasus bola dengan pusat sepanjang sumbu *Y*, persamaan bola dinyatakan sebagai berikut

$B(\emptyset,\theta) =< r.\sin\emptyset.\cos\theta + 0, r.\sin\emptyset.\sin\theta + y, r.\cos\emptyset + 0 >$ (2.10)

dan persamaan parametrik bola dengan sumbu pusat *X*, yaitu

$B(\emptyset,\theta) =< r.\sin\emptyset.\cos\theta + x, r.\sin\emptyset.\sin\theta + 0, r.\cos\emptyset + 0 >$ (2.11)

Jika menginginkan suatu potongan bola dengan pusat $Q(a,b,c)$ yang dipotong tegak lurus terhadap sumbu *x*, *y* atau *z*, maka potongan bola dapat ditentukan melalui persamaan (2.9), (2.10) dan (2.11) dengan parameter $0 \leq \theta \leq 2\pi$ dan $\emptyset_{min} \leq \emptyset \leq \emptyset_{max}$ serta

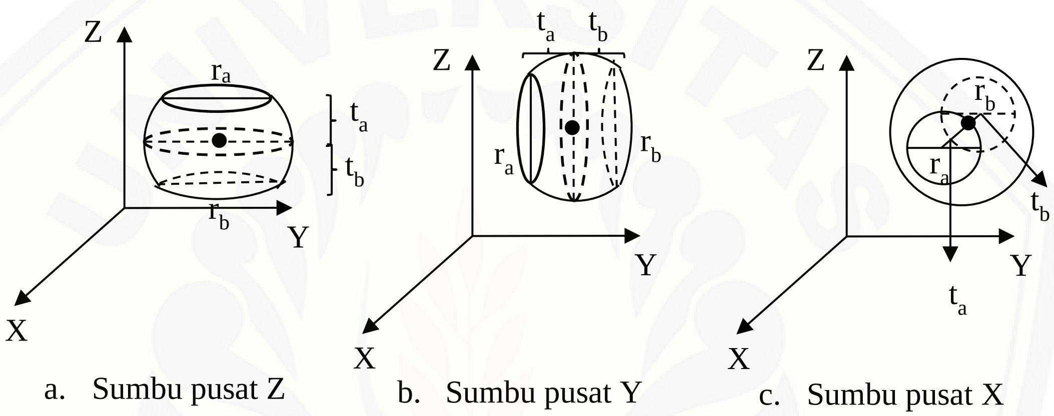
jari-jari alas pertama = $r_a = r.\sin(\emptyset_{min})$,

jari-jari alas kedua = $r_b = r.\sin(\emptyset_{max})$,

tinggi alas pertama = $t_a = r.\cos(\emptyset_{min})$,

tinggi alas kedua = $t_b = r.\cos(\emptyset_{max})$.

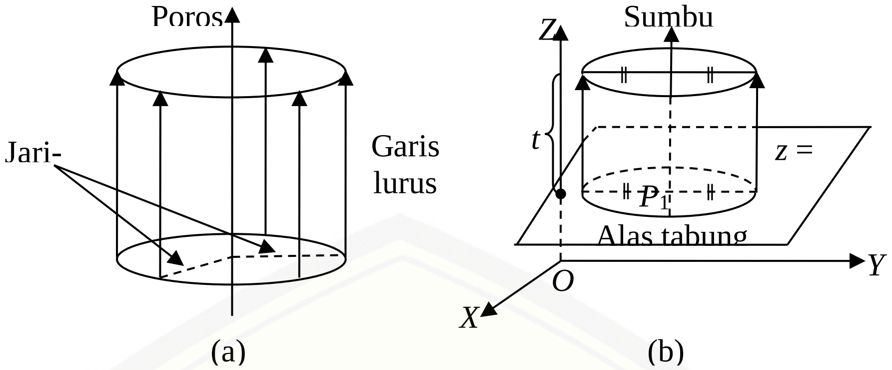
Hasil dari bentuk keratan bola dengan sumbu pusat Z, Y, dan X masing-masing ditunjukkan pada Gambar 2.9 a,b,c.




Gambar 2.9 Keratan bola dengan pusat *Q(a,b,c)*

### 2.3.2 Penyajian Tabung

Tabung dibangun oleh garis lurus yang sejajar dengan garis lurus tertentu (poros) yang bergerak sejajar dengan jari-jari yang bersifat konstan. Tabung juga dapat diartikan sebagai silinder lingkaran tegak yang merupakan tempat kedudukan garis-garis sejajar tegak dan berjarak sama terhadap garis tertentu (Suryadi,1986) (Gambar 2.10).




Gambar 2.10 Penyajian Tabung

Menurut Bastian (2011), jika diketahui tabung dengan pusat alas $P_1(x_1, y_1, z_1)$ jari-jari $R$ dan tinggi $t$, maka dapat dicari persamaan parametrik tabung sebagai berikut.

a. Jika alas terletak pada bidang $z = z_1$ dan sumbu pusat tabung sejajar sumbu $z$, maka untuk mencari persamaan parametrik tabung dapat dilakukan dengan langkah-langkah sebagai berikut (Gambar 2.11a).

   1. Tentukan persamaan parametrik lingkaran dengan pusat $P_1(x_1, y_1, z_1)$, jari-jari $R$ dan terletak pada bidang $z = z_1$ yaitu

   $$\boldsymbol{L}(\theta) = \langle x_1 + R\cos\theta,\ y_1 + R\sin\theta, z_1\rangle \quad (2.12)$$

   dengan $0 \leq \theta \leq 2\pi$ dan $R$ bilangan real.

   2. Translasikan lingkaran (2.12) dari $z_1$ sampai $z_1 + t$ sehingga terbentuk persamaan parametrik tabung.

   $$\boldsymbol{T}(\theta, z) = \langle x_1 + R\cos\theta, y_1 + R\sin\theta, z\rangle, \quad (2.13)$$

   dengan $0 \leq \theta \leq 2\pi$ dan $z_1 \leq z \leq z_1 + t$.

b. Jika alas terletak pada bidang $x = x_1$ dan sumbu pusat tabung sejajar sumbu $X$, maka untuk mencari persamaan parametrik tabung dapat dilakukan dengan mengulangi langkah a dan didapatkan persamaan (Gambar 2.11b)

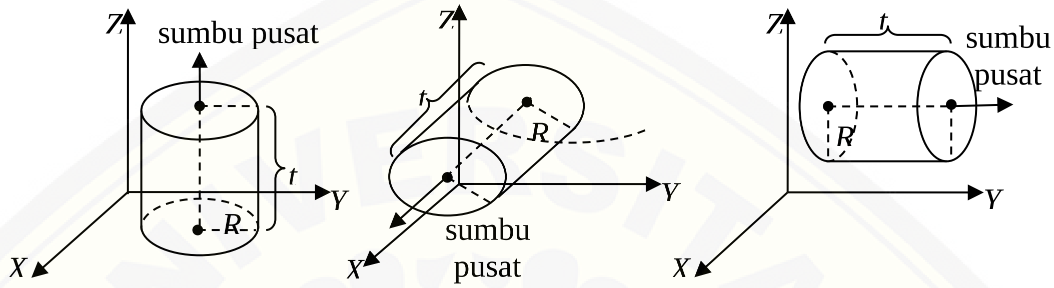
$$\boldsymbol{T}(\theta, z) = \langle x, y_1 + R\sin\theta, z_1 + R\cos\theta\rangle, \quad (2.14)$$

dengan $0 \leq \theta \leq 2\pi$ dan $x_1 \leq x \leq x_1 + t$.

c. Jika alas terletak pada bidang $y = y_1$ dan sumbu pusat tabung sejajar sumbu $Y$, maka untuk mencari persamaan parametrik tabung dapat dilakukan dengan mengulangi langkah a dan b sehingga diperoleh persamaan (Gambar 2.11c)

$$\boldsymbol{T}(\theta,z) = \langle x_1 + R\cos\theta,\ y,\ z_1 + R\sin\theta\rangle, \tag{2.15}$$

dengan $0 \le \theta \le 2\pi$ dan $y_1 \le y \le y_1 + t$.

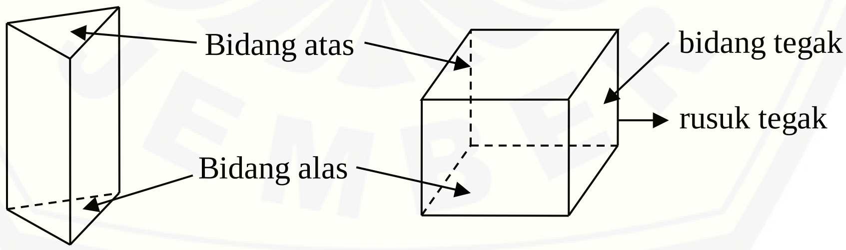



(a) Sumbu Pusat Sejajar $Z$ (b) Sumbu Pusat Sejajar $X$ (c) Sumbu Pusat Sejajar $Y$

Gambar 2.11 Tabung dengan beragam sumbu pusat

### 2.3.3 Penyajian Prisma Segienam Beraturan

Prisma adalah suatu benda ruang tertutup yang dibatasi oleh dua bidang sejajar dan beberapa bidang datar perpotongan dengan garis-garis potong sejajar. Dua bidang yang sejajar tersebut dinamakan bidang alas dan bidang atas, bidang-bidang perpotongan disebut dengan bidang tegak, sedangkan jarak antara bidang alas dan bidang atas disebut tinggi prisma (Gambar 2.12).




Gambar 2.12 Prisma dan Bagiannya

Penamaan prisma diambil dari nama poligon yang menjadi bidang alas dan bidang atasnya. Jika bidang alas dan bidang atas berbentuk segienam, maka prisma tersebut disebut prisma segienam.

Misalkan diketahui segienam beraturan $K_1K_2K_3K_4K_5K_6$ dengan koordinat titik-titik sudut $K_1(x_1, y_1, z_1)$, $K_2(x_2, y_2, z_2)$, $K_3(x_3, y_3, z_3)$, $K_4(x_4, y_4, z_4)$,

$K_5(x_5, y_5, z_5)$ dan $K_6(x_6, y_6, z_6)$ sebagai alas prisma. Dari data titik-titik tersebut dapat dikonstruksi prisma segienam beraturan dengan langkah-langkah sebagai berikut.

1. Menetapkan tiga titik $K_1$, $K_2$, $K_3$ dan vektor $\overrightarrow{K_1K_2}, \overrightarrow{K_3K_2}$ dengan

$$\overrightarrow{K_1K_2} = < x_2 - x_1, y_2 - y_1, z_2 - z_1 >,$$
$$\overrightarrow{K_3K_2} = < x_2 - x_3, y_2 - y_3, z_2 - z_3 >.$$

2. Menghitung vektor normal bidang ($\boldsymbol{n}_{\alpha_u}$) alas menggunakan persamaan

$$\boldsymbol{n}_{\alpha_u} = \langle \frac{a}{\sqrt{a^2+b^2+c^2}}, \frac{b}{\sqrt{a^2+b^2+c^2}}, \frac{c}{\sqrt{a^2+b^2+c^2}} \rangle = \langle a_1, a_2, a_3 \rangle,$$

dengan

$$a = y_1(z_3 - z_2) + y_2(z_1 - z_3) + y_3(z_2 - z_1),$$
$$b = x_1(z_2 - z_3) + x_2(z_3 - z_1) + x_3(z_1 - z_2),$$
$$c = x_1(y_3 - y_2) + x_2(y_1 - y_3) + x_3(y_2 - y_1).$$

3. Mentranslasikan alas prisma dengan tinggi $t$ sejajar $\boldsymbol{n}_{\alpha_u} = \langle a_1, a_2, a_3 \rangle$ sehingga didapatkan bidang atas prisma dengan titik sudut $K_1'$, $K_2'$, $K_3'$, $K_4'$, $K_5'$ dan $K_6'$ sehingga diperoleh:

$$\overrightarrow{OK_1}' = \overrightarrow{OK_1}' + t\boldsymbol{n}_{\alpha_u} \Rightarrow \overrightarrow{OK_1}' = \begin{pmatrix} x_1 \\ y_1 \\ z_1 \end{pmatrix} + t\begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix},$$

$$\overrightarrow{OK_2}' = \overrightarrow{OK_2}' + t\boldsymbol{n}_{\alpha_u} \Rightarrow \overrightarrow{OK_2}' = \begin{pmatrix} x_2 \\ y_2 \\ z_2 \end{pmatrix} + t\begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix},$$

$$\overrightarrow{OK_3}' = \overrightarrow{OK_3}' + t\boldsymbol{n}_{\alpha_u} \Rightarrow \overrightarrow{OK_3}' = \begin{pmatrix} x_3 \\ y_3 \\ z_3 \end{pmatrix} + t\begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix},$$

$$\overrightarrow{OK_4}' = \overrightarrow{OK_4}' + t\boldsymbol{n}_{\alpha_u} \Rightarrow \overrightarrow{OK_4}' = \begin{pmatrix} x_4 \\ y_4 \\ z_4 \end{pmatrix} + t\begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix},$$

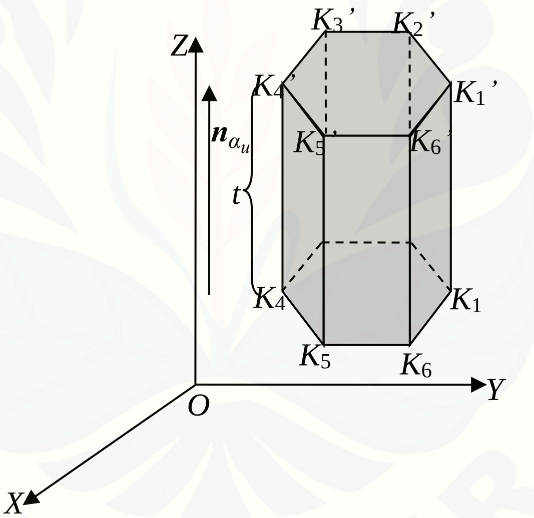
$$\overrightarrow{OK_5}' = \overrightarrow{OK_5}' + t\boldsymbol{n}_{\alpha_u} \Rightarrow \overrightarrow{OK_5}' = \begin{pmatrix} x_5 \\ y_5 \\ z_5 \end{pmatrix} + t\begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix},$$

$$\overrightarrow{OK_6}' = \overrightarrow{OK_6}' + t\boldsymbol{n}_{\alpha_u} \Rightarrow \overrightarrow{OK_6}' = \begin{pmatrix} x_6 \\ y_6 \\ z_6 \end{pmatrix} + t\begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix}.$$

4. Menginterpolasi segmen-segmen garis pada bidang alas dan bidang atas prisma menggunakan Persamaan (2.8) sehingga diperoleh enam bidang segienam dengan persamaan

$$S_{K_1K_2K_1'K_2'}(u,v) = (1-v)\overline{K_1K_2}(u) + v\overline{K_1'K_2'}(u),$$
$$S_{K_2K_3K_2'K_3'}(u,v) = (1-v)\overline{K_2K_3}(u) + v\overline{K_2'K_3'}(u),$$
$$S_{K_3K_4K_3'K_4'}(u,v) = (1-v)\overline{K_3K_4}(u) + v\overline{K_3'K_4'}(u),$$
$$S_{K_4K_5K_4'K_5'}(u,v) = (1-v)\overline{K_4K_5}(u) + v\overline{K_4'K_5'}(u),$$
$$S_{K_5K_6K_5'K_6'}(u,v) = (1-v)\overline{K_5K_6}(u) + v\overline{K_5'K_6'}(u),$$
$$S_{K_1K_6K_1'K_6'}(u,v) = (1-v)\overline{K_1K_6}(u) + v\overline{K_1'K_6'}(u).$$

dengan $0 \leq u \leq 1$ dan $0 \leq v \leq 1$.




Gambar 2.13 Penyajian prisma segienam beraturan

### 2.4 Transformasi Bidang di $R^3$

Transformasi bidang di $R^3$ ini terdiri dari translasi, rotasi (perputaran), dilatasi (penskalaan) dan refleksi (pencerminan).

2.4.1 Translasi

Translasi sebagai elemen dasar, setiap titik di $R^3$ ditentukan oleh tiga referensi, yaitu ke arah sumbu $X$, ke arah sumbu $Y$ dan ke arah sumbu $Z$. Sebarang titik $Q$ dinyatakan sebagai $(X_q, Y_q, Z_q)$ dalam bentuk koordinat dan $< X_q, Y_q, Z_q >$

dalam bentuk vektor. Translasi adalah perpindahan kedudukan sebarang titik dengan penambahan/pengurangan besaran pada arah sumbu $X$, $Y$ dan $Z$. (Budhi, 1995)

Secara umum, translasi dapat dinyatakan oleh persamaan $Q = TP + K$, dimana $P$ adalah posisi awal, $Q$ adalah posisi titik setelah ditranslasi, $T$ adalah matriks identitas dan $K$ menunjukkan besarnya pergeseran ke arah sumbu $X$, $Y$ dan $Z$.

Hasil translasi dapat dinyatakan sebagai:

$$\left(X_q, Y_q, Z_q\right) = (X_p + K_x, Y_p + K_y, Z_p + K_z)$$

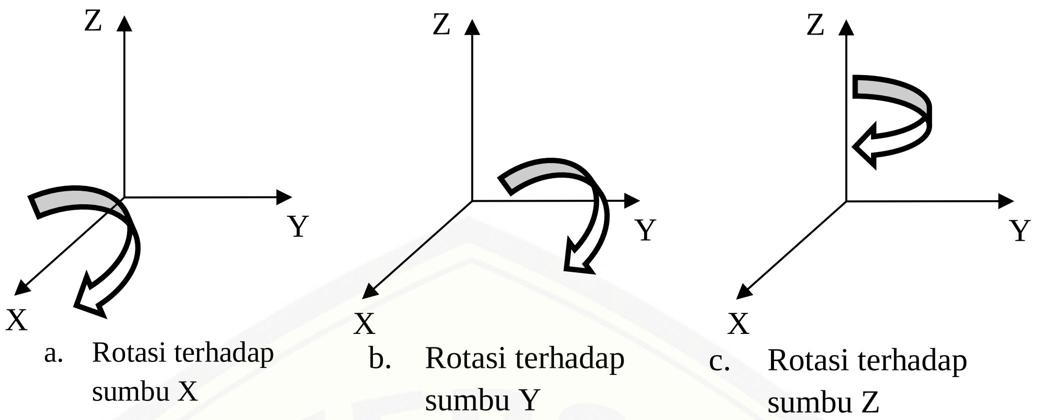
Dalam bentuk matriks, notasi diatas dapat ditulis sebagai berikut

$$\begin{bmatrix} X_q \\ Y_q \\ Z_q \end{bmatrix} = \overset{A_2}{\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}} \begin{bmatrix} X_p \\ Y_p \\ Z_p \end{bmatrix} + \begin{bmatrix} K_x \\ K_y \\ K_z \end{bmatrix} \tag{2.16}$$

Matriks $A_2$ merupakan matriks koefisien yang bersesuaian dengan transformasi translasi. Translasi bersifat mempertahankan bentuk dan ukuran obyek.

### 2.4.2 Rotasi (Perputaran)

Dalam $R^3$ dikenal dua sistem koordinat, yaitu sistem koordinat tangan kanan dan sistem koordinat tangan kiri. Pada sistem koordinat tangan kiri, rotasi bersudut positif dinyatakan searah dengan arah putaran jarum jam. Sedangkan, pada sistem koordinat tangan kanan, rotasi yang bersudut positif dinyatakan sebagai berlawanan arah dengan arah putaran jarum jam.




Gambar 2.14 Ilustrasi rotasi pada sistem koordinat tangan kiri

Apabila $\theta$ menunjukkan besarnya sudut rotasi dengan titik pangkal rotasi O(0,0,0), maka rotasi terhadap masing-masing sumbu dapat dituliskan dalam bentuk matriks sebagai berikut

Rotasi terhadap sumbu x:

$$\begin{bmatrix} X_q \\ Y_q \\ Z_q \end{bmatrix} = \overbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & \sin\theta \\ 0 & -\sin\theta & \cos\theta \end{bmatrix}}^{B_1} \begin{bmatrix} X_p \\ Y_p \\ Z_p \end{bmatrix} \tag{2.17}$$

Rotasi terhadap sumbu y:

$$\begin{bmatrix} X_q \\ Y_q \\ Z_q \end{bmatrix} = \overbrace{\begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix}}^{B_2} \begin{bmatrix} X_p \\ Y_p \\ Z_p \end{bmatrix} \tag{2.18}$$

Rotasi terhadap sumbu z:

$$\begin{bmatrix} X_q \\ Y_q \\ Z_q \end{bmatrix} = \overbrace{\begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}}^{B_3} \begin{bmatrix} X_p \\ Y_p \\ Z_p \end{bmatrix} \tag{2.19}$$

Dalam hal ini, matriks B1, B2 dan B3 merupakan matriks koefisien yang bersesuaian dengan transformasi rotasi. Rotasi memiliki sifat yang sama dengan translasi.

2.4.3 Dilatasi (Penskalaan)

Dilatasi atau penskalaan adalah transformasi yang mengubah jarak titik-titik dengan faktor pengali tertentu $(k)$ terhadap suatu titik tertentu yang disebut sebagai pusat dilatasi. Oleh karena itu, dilatasi juga dapat digunakan untuk mengubah ukuran benda dengan memperbesar atau memperkecil ukuran awal. Menurut Kusno (2009), transformasi dilatasi yang memetakan titik $P(x, y, z)$ ke $P'(x', y', z')$ didefinisiskan dengan bentuk formula berikut:

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} k_1 & 0 & 0 \\ 0 & k_2 & 0 \\ 0 & 0 & k_3 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \begin{bmatrix} k_1 x \\ k_2 x \\ k_3 x \end{bmatrix} \qquad (2.20)$$

dengan $k_1, k_2, k_3$ bilangan-bilangan real.

Dalam hal ini pemilihan harga $k_1$ menyajikan skala ke arah sumbu $X$, $k_2$ ke arah sumbu $Y$ dan $k_3$ menyajikan skala ke arah sumbu $Z$, jika $k_1 = k_2 = k_3$, maka peta obyek yang didapat sebangun dengan obyek aslinya (mungkin diperbesar, diperkecil atau tetap).

## 2.5 Penyajian Kurva dan Permukaan Bezier

Menurut Kusno (2009), kurva Bezier derajat-$n$ C($u$) dinyatakan dalam bentuk parametrik berikut:

$$\mathrm{C}(u) = \sum_{i=0}^{n} \mathbf{P}_i \mathrm{B}_i^n(u),\ 0 \le u \le 1 \qquad (2.21)$$

dengan:

$\mathrm{B}_i^n(u) = \mathrm{C}_i^n (1-u)^{n-i} u^i$,

$\mathrm{C}_i^n = \frac{n!}{i!(n-i)!}$,

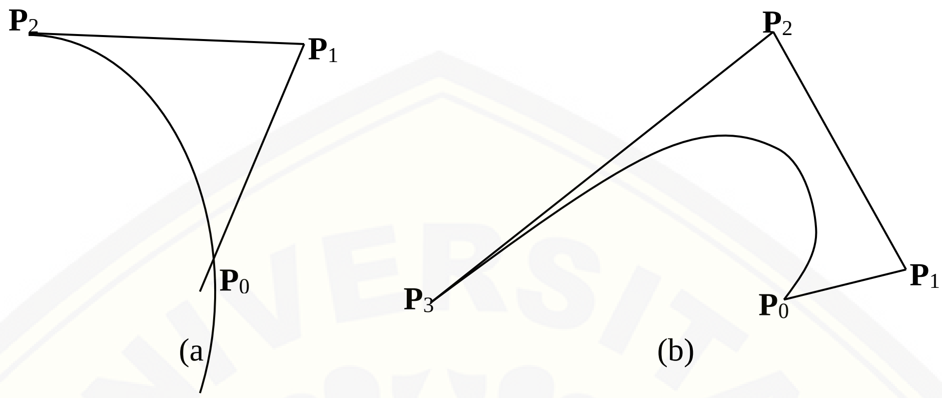
$\mathbf{P}_i$ = koefisien geometri / titik kontrol kurva C($u$).

Jika $n$ = 2, akan dihasilkan kurva Bezier kuadratik dengan persamaan parametrik (Gambar 2.13a):

$$\mathrm{C}(u) = (1-u)^2 \mathbf{P}_0 + 2(1-u)\, u\, \mathbf{P}_1 + u^2 \mathbf{P}_2,$$

sedangkan untuk $n$ = 3 didapatkan empat titik kontrol yaitu $\mathbf{P}_0$, $\mathbf{P}_1$, $\mathbf{P}_2$, dan $\mathbf{P}_3$ sehingga persamaan parametrik kurva Bezier kubiknya adalah (Gambar 2.13b):

$$C(u) = (1 - u)^3\mathbf{P}_0 + 3(1 - u)^2u\,\mathbf{P}_1 + 3(1 - u)u^2\,\mathbf{P}_2 + u^3\,\mathbf{P}_3.$$




Gambar 2.15 Kurva Bezier (a) kuadratik (b) kubik

Permukaan Bezier pada prinsipnya identik dengan kurva Bezier. Permukaan Bezier S($u$,$v$) derajat $m$ dan $n$ dinyatakan dalam bentuk parametrik berikut (Gambar 2.16):

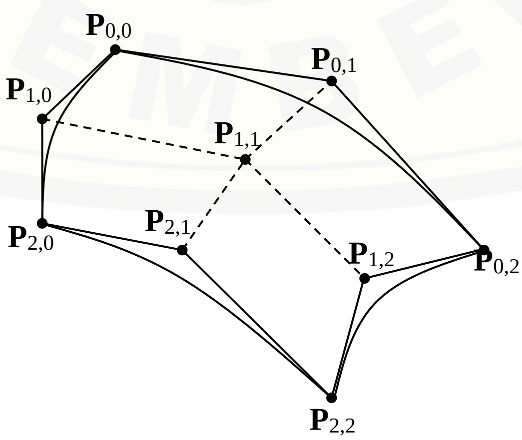
$$S(u,v) = \sum_{i,j=0}^{m,n} \mathbf{P}_{ij}B_i^m(u)B_j^n(v),\ 0 \leq u,v \leq 1 \tag{2.22}$$

dengan:

$B_i^m(u) = \frac{m!}{i!(m-i)!}(1-u)^{m-i}u^i$,

$B_j^n(v) = \frac{n!}{j!(n-j)!}(1-v)^{n-i}v^j$,

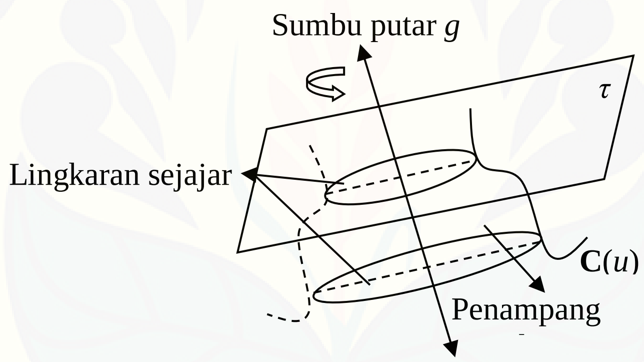
$\mathbf{P}_{ij}$ = koefisien geometri / titik kontrol permukaan $\mathbf{S}(u,v)$.




Gambar 2.16 Permukaan Bezier dengan $m$ = 2 dan $n$ = 2

### 2.6 Permukaan Putar

Menurut Kusno (2009), permukaan putar adalah suatu permukaan yang dibangkitkan oleh suatu kurva ruang $\mathbf{C}(u)$ (sebagai generatrik) diputar mengitari sebuah sumbu putar $g$ yang disebut sebagai sumbu putar (Gambar 2.17).
Dalam membahas permukaan putar, terdapat beberapa istilah yang perlu diketahui. Pertama, bagian-bagian bidang penampang yang melalui sumbu putar dan dibatasi oleh permukaan putar, disebut dengan istilah penampang-penampang meridian. Semua penampang-penampang meridian adalah saling kongruen. Sedangkan lingkaran-lingkaran sejajar permukaan putar adalah perpotongan antara bidang-bidang sejajar yang tegak lurus sumbu putar dengan permukaan putar.




Gambar 2.17 Permukaan putar

Misalkan $C_x(u)$, $C_y(u)$, dan $C_z(u)$ menyatakan komponen-komponen skalar dari kurva generatris $\boldsymbol{C}(u)$, maka permukaan putar yang dibangkitkan oleh kurva $\boldsymbol{C}(u)$ dapat diformulasikan sebagai berikut.

a. Jika kurva generatris $\boldsymbol{C}(u)$ pada bidang $YOZ$ dan sumbu putar $OZ$, maka untuk mencari persamaan parametrik permukaan putar dilakukan dengan langkah-langkah sebagai berikut (Gambar 2.18a).
   1. Tentukan persamaan parametrik kurva $\boldsymbol{C}(u)$, yaitu
   $$\boldsymbol{C}(u) = < C_x(u), C_y(u), C_z(u) > \quad (2.23)$$
   dengan $0 \leq u \leq 1$.

2. Putar kurva $\boldsymbol{C}(u)$ terhadap sumbu putar $OZ$, maka terbentuk sebuah permukaan putar dengan persamaan parametrik

$$\boldsymbol{S}(u,v) = <\ C_x(u)\cos v,\ C_y(u)\sin v,\ C_z(u) > \tag{2.24}$$

dengan $0 \leq u \leq 1$ dan $0 \leq v \leq 2\pi$.

b. Jika kurva generatris $\boldsymbol{C}(u)$ pada bidang $XOZ$ dan sumbu putar $OY$, maka untuk mencari persamaan parametrik permukaan putar dilakukan dengan mengulangi langkah a dan didapatkan persamaan (Gambar 2.18b)

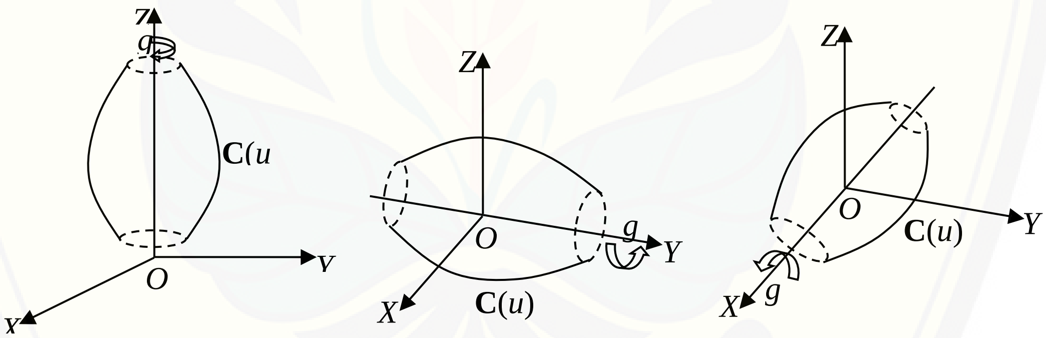
$$\boldsymbol{S}(u,v) = <\ C_x(u)\cos v,\ C_y(u),\ C_z(u)\sin v > \tag{2.25}$$

dengan $0 \leq u \leq 1$ dan $0 \leq v \leq 2\pi$.

c. Jika kurva generatris $\boldsymbol{C}(u)$ pada bidang $XOY$ dan sumbu putar $OX$, maka untuk mencari persamaan parametrik permukaan putar dilakukan dengan mengulangi langkah a dan didapatkan persamaan (Gambar 2.18c)

$$\boldsymbol{S}(u,v) = <\ C_x(u), C_y(u)\ \cos v\ ,\ C_z(u)\sin v > \tag{2.26}$$

dengan $0 \leq u \leq 1$ dan $0 \leq v \leq 2\pi$.




(a) sumbu putar (b) sumbu putar (c) sumbu putar

Gambar 2.18 Permukaan putar kurva $\boldsymbol{C}(u)$

## 2.7 Teknik Deformasi

Deformasi benda geometri adalah mengubah bentuk benda awal menjadi benda baru dengan memberikan perlakuan atau metode yang bersesuaian. Perubahan tersebut meliputi perubahan bentuk atau ukuran. Beberapa metode deformasi yang dimaksud yaitu berupa pemotongan (interseksi), memutar kurva, interpolasi, transformasi dilatasi dan pemuntiran. Deformasi dibagi menjadi dua yaitu deformasi sebagian dan deformasi total. Deformasi sebagian dilakukan dengan mengubah ukuran benda namun bentuk yang dihasilkan masih sama sehingga bentuk yang dihasilkan sebangun. Deformasi total adalah mengubah tampilan atau bentuk benda secara menyeluruh atau total sehingga hasil yang diperoleh tidak sebangun dari benda awal.

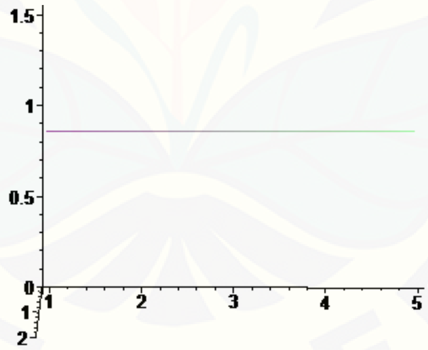
## 2.8 Konstruksi Objek pada Program Maple 18

Pada subbab ini disajikan beberapa contoh konstruksi objek-objek geometri dengan *software* Maple 18 untuk mengkonstruksi objek geometri.

a. Penyajian Segmen Garis

Untuk membuat segmen garis menggunakan maple, dapat menggunakan Persamaan (2.1c) dengan memberikan nilai $(x_1, y_1, z_1)$ dan $(x_2, y_2, z_2)$ sebagai posisi titik ujung segmen garis di ruang. Misalkan akan dibuat suatu segmen garis $a$ (Gambar 2.19) dengan titik-titik ujung $A$(0,0,0) dan $B$(1,15,1). Berikut ini merupakan *script* program Maple 18.

```
a1:=spacecurve([(1-t)*0+t*0,(1-t)*0+t*15,(1-t)*0+t*1],
t=0..1):
```

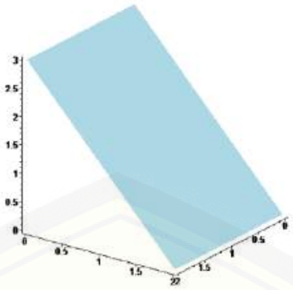



Gambar 2.19 Segmen garis

b. Penyajian Bidang Segiempat

Misalkan dibangun bidang segiempat (Gambar 2.20) dengan titik sudut-titik sudut $A$(2,2,0), $B$(0,2,0), $C$(2,0,3) dan $D$(0,0,3) maka bentuk perintahnya sebagai berikut.

```
a2:=plot3d([(1-v)*(2-2*u)+v*(2-2*u),(1-v)*2+v*0,(1-
v)*0+v*3],u=0..1,v=0..1):
```

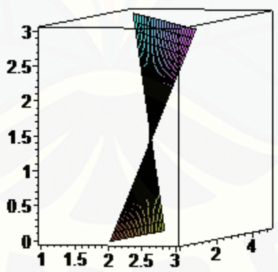


Gambar 2.20 Bidang segiempat

c. Penyajian Permukaan Tidak Datar

Sama halnya dengan penyajian bidang segitiga dan segiempat, untuk membuat permukaan tidak datar juga dapat menggunakan Persamaan (2.7), hanya kurva batasnya dipilih yang menyilang satu sama lain. Dibuat bidang atau permukaan tidak datar *c* dari titik-titik *A*(2,0,0), *B*(2,3,0), *C*(3,1,3) dan *D*(-1,5,3). Hasilnya dapat disajikan pada (Gambar 2.21) dengan *script* sebagai berikut.

```
a3:=plot3d([(1-v)*2+v*(3-2*u),(1-v)*3*u+v*(1+4*u),(1-v)
*0+v*3],u=0..1,v=0..1):
```

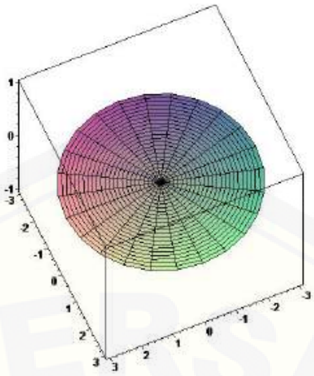



Gambar 2.21 Bidang permukaan tidak datar

d. Penyajian Bidang Lingkaran

Untuk membuat bidang lingkaran dapat menggunakan Persamaan (2.3) dengan memberikan nilai jari-jari dan titik pusatnya. Misalkan akan dibentuk lingkaran *e* (Gambar 2.22) dengan pusat di *A*(0,0,0) dan jari-jari sepanjang 3 satuan. Berikut ini contoh *script*-nya.

```
a4:=plot3d([r*3*cos(t)+0,r*3*sin(t)+0,0],r=0..1,t=0..2*
Pi):
```

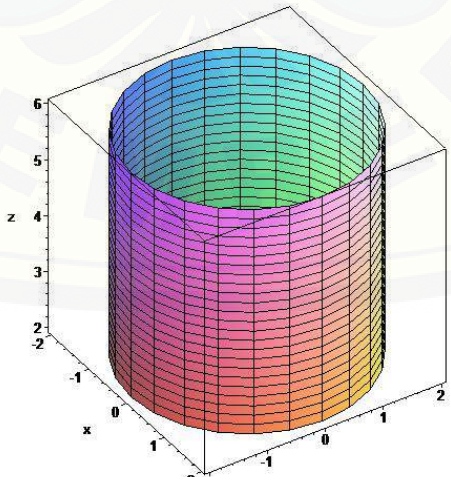


Gambar 2.22 Bidang lingkaran

e. Penyajian Permukaan Selimut Silinder

Untuk membuat permukaan selimut silinder dapat menggunakan Persamaan (2.15) dengan memberikan nilai jari-jari, titik pusat dan ketinggian silinder. Misalkan akan dibentuk silinder *h* (Gambar 2.20) dengan pusat di *A*(0,0,0), jari-jari sepanjang 2 satuan dan ketinggian 6 satuan. Berikut ini contoh *scrip*-nya.

```
a5:=plot3d([2*cos(u)+0,2*sin(u)+0,2*v],u=0..2*Pi,v=1..3
,scaling=constrained,labels=([x,y,z]):
```

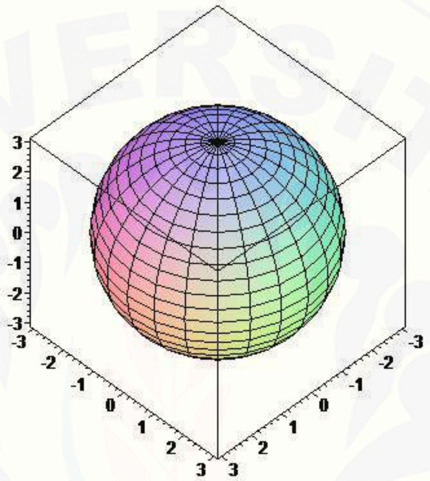



Gambar 2.23 Permukaan Selimut Silinder

f. Penyajian Permukaan Bola

Untuk membuat permukaan bola dapat menggunakan Persamaan (2.9) dengan memberikan nilai jari-jari dan titik pusat bola. Misalkan akan dibentuk bola (Gambar 2.24) dengan pusat di *A*(0,0,0) dan jari-jari sepanjang 3 satuan. Berikut ini contoh *script*.

```
a6:=plot3d([3*sin(v)*cos(u)+0,3*sin(v)*sin(u)+0,3*cos(v
)+0],u=0..2*Pi,v=0..Pi,scaling=constrained):
```



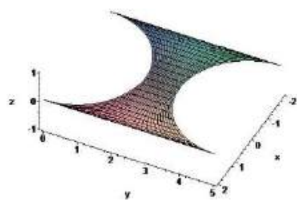

Gambar 2.24 Permukaan Bola

g. Penyajian interpolasi antara dua kurva

Misalkan akan menginterpolasi antara dua kurva yang diberi nama *ll* dengan kurva pertama berupa setengah lingkaran berpusat di (0,0,0) sedangkan kurva kedua berupa lingkaran berpusat di (0,5,0) dengan jari-jari masing-masing 2 satuan. Berikut ini merupakan contoh *script*-nya:

```
ll:=plot3d([(1-v)*2*cos(t)+v*(2*cos(-t)),(1-
v)*2*sin(t)+ v*(2*sin(-t)+5),0],v=0..1,t=0..Pi):
```

Permukaan hasil interpolasi ditunjukkan pada Gambar 2.25.

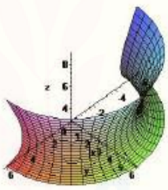


Gambar 2.25 Interpolasi antara dua kurva

h. Penyajian permukaan Bezier

Pada program Maple 18 untuk membangun permukaan Bezier misalnya permukaan Bezier *g*, seperti ditunjukkan pada Gambar 2.26 dapat dituliskan contoh *script* program sebagai berikut.

```
b:=([4*(1-t)^2+5*2*(1-t)*t+t^2*0,0*(1-t)^2+0*2*(1-
t)*t+t^2*0,8*(1-t)^2+7*2*(1-
t)*t+t^2*7],t=0..1,color=red,thickness=5,labels=[x,y,z]
):
z5:=spacecurve(p):display(z5):
c:=plot3d([(6*(1-t)^2+3*2*(1-t)*t+t^2*6)*cos(v),(6*(1-
t)^2+3*2*(1-t)*t+t^2*6)*sin(v),(9*(1-t)^2+3*2*(1-
t)*t+t^2*3)],t=0..1,v=0..Pi,labels=[x,y,z],axes=normal,
scaling=constrained):
```



Gambar 2.26 Permukaan Bezier

# BAB 3. METODE PENELITIAN

Berdasarkan rumusan masalah pada subbab 1.2 dan hasil kajian tinjauan pustaka pada Bab 2, maka berikut akan diuraikan beberapa metode penelitian untuk menyelesaikan permasalahan. Pertama menentukan data berupa prisma segienam beraturan, tabung, dan bola. Kedua memodelisasi prisma segienam beraturan, tabung dan bola untuk menyusun bagian-bagian tiang teras. Ketiga penggabungan hasil modelisasi prisma segienam beraturan, tabung, dan bola untuk mendapatkan hasil model bentuk tiang teras. Terakhir menyusun program dan visualisasi komputer menggunakan maple 18. Untuk lebih jelas mengenai metode penelitian tersebut akan diuraikan sebagai berikut

## 3.1 Data

Data berupa prisma segienam beraturan, tabung dan bola masing-masing akan ditetapkan sebagai berikut

1. Prisma segienam beraturan tegak dengan alas $20 \leq a \leq 40$ satuan dari titik berat ke titik sudut dan tinggi $20 \leq t \leq 80$ satuan,
2. Tabung dengan selimut tegak dengan jari-jari $20 \leq r \leq 40$ satuan dan tinggi $40 \leq t \leq 80$ satuan, dan
3. Bola dengan titik pusat $Q(a, b, c)$ dan jari-jari $20 \leq r \leq 40$ satuan.

## 3.2 Metode

Metode yang digunakan pada penelitian ini adalah teknik deformasi untuk memodelisasi tiap benda dasar penyusun tiang teras sebagai berikut

1. Interpolasi

   Pada metode interpolasi digunakan pada deformasi prisma segienam efek puntiran, potong pojok dan memberikan lengkung pada sisi tegak. Untuk deformasi tabung interpolasi digunakan pada memberikan lengkung pada selimut tabung dan pemotongan menggunakan pola tutup tabung. Untuk deformasi bola interpolasi digunakan untuk memberikan keratan pada kulit bola.

2. Dilatasi

   Metode deformasi dengan menggunakan transformasi dilatasi digunakan pada deformasi bola dengan memberikan keratan pada kulit bola.
3. Memutar kurva

   Metode deformasi memutar kurva digunakan pada deformasi tabung.
4. Interseksi

   Metode deformasi interseksi digunakan pada deformasi prisma segienam, tabung dan bola.
5. Memuntir

   Metode deformasi memuntir digunakan pada deformasi prisma segienam.

## 3.3 Modelisasi benda dasar

Tahap berikutnya merupakan langkah-langkah untuk modelisasi prisma segienam beraturan, tabung dan bola

### 3.3.1 Modelisasi Prisma Segienam

Kasus modelisasi prisma segienam diperoleh dengan langkah-langkah sebagai berikut

a) Memberikan sudut sebesar $\theta$ pada tutup atas prisma kemudian merotasi tutup atas prisma sebesar $\theta$.
b) Memberikan 3 titik pada sisi tegak prisma kemudian memberikan lengkung berupa lengkung cembung $(Q > a)$ dan lengkung cekung $(Q < a)$ pada sisi tegak prisma, dengan $Q$ merupakan titik kontrol kurva untuk membentuk lengkungan.
c) Membangun bentuk permukaan pada bagian sisi tegak prisma maupun pada hasil pemotongan.

### 3.3.2 Modelisasi Tabung

Kasus modelisasi tabung diperoleh dengan langkah-langkah sebagai berikut

a) Membagi tabung menjadi 3 dan 4 bagian secara vertikal agar diperoleh variasi ukuran dan permukaan lengkung pada selimut tabung.

b) Memberikan pola tutup tabung dengan menetapkan banyaknya busur pada tutup tabung.

### 3.3.3 Modelisasi Bola

Kasus modelisasi bola diperoleh dengan langkah-langkah sebagai berikut

a) Memberikan titik pada jari-jari bola sejajar sumbu z, kemudian menetapkan bidang datar pada titik tersebut untuk pemotongan.
b) Mendilatasi jari-jari bola agar bola memiliki kulit yang bervariasi.
c) Memberikan efek keratan pada kulit bola.

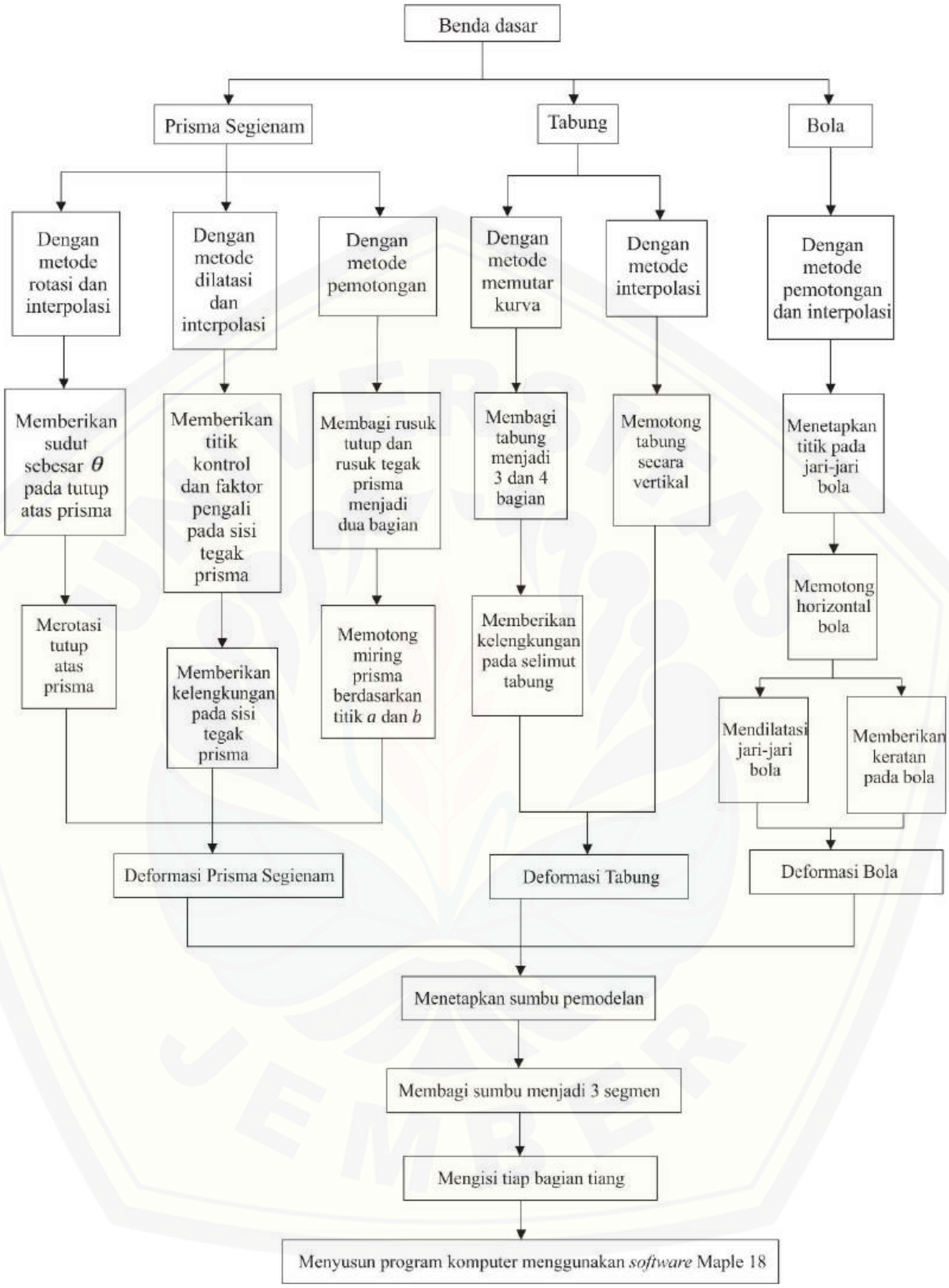
## 3.4 Penggabungan Hasil Modelisasi

Penggabungan hasil modelisasi prisma segienam beraturan,tabung dan bola untuk mendapatkan bentuk model tiang teras, dapat diuraikan sebagai berikut

1. Membangun sumbu pemodelan untuk merangkai benda yang terbentuk melalui hasil modelisasi prisma segienam beraturan,tabung dan bola.
2. Mengidentifikasi bentuk benda yang memiliki bentuk dan ukuran sambungan yang sama sehingga dapat dilakukan penggabungan bagian-bagian masing-masing komponen dari benda dasar yang dimodelisasi.

## 3.5 Penyusunan Program

Menyusun program dan visualisasi komputer hasil analisis (a), (b) dan (c) menggunakan *software* Maple 18. Untuk lebih jelas mengenai metode penelitian tersebut dapat dilihat pada skema (Gambar 3.1) berikut ini:




Gambar 3.1 Skema metode penelitian

# BAB 5. KESIMPULAN DAN SARAN

## 5.1 Kesimpulan

Berdasarkan hasil penelitian dan pembahasan pada bab 4, didapatkan bahwa untuk memodelisasi tiang teras perlu dilakukan langkah-langkah sebagai berikut.

a. Untuk mendesain beragam bentuk benda dasar penyusun tiang teras dari benda prisma segienam, tabung, dan bola dapat dilakukan prosedur sebagai berikut. Pertama menetapkan bidang yang akan dijadikan bidang potong untuk prisma segienam, tabung, dan bola kemudian menggunakan bidang-bidang tersebut untuk memotong benda-benda geometri sehingga menghasilkan benda-benda geometri terpancung. Kedua, menetapkan titik-titik atau pola pada tiap sisi tegak atau tutup prisma, tabung dan bola. Ketiga, mengoperasikan titik-titik tersebut yaitu: (a) menetapkan titik kontrol untuk memperkecil atau memperbesar jari-jari dan ketinggian, (b) membangun segmen garis, bidang lingkaran, kurva bezier kuadratik. Terakhir menginterpolasikan kurva atau memutar kurva sehingga menghasilkan bentuk komponen tiang teras.
b. Untuk merangkai bagian tiang teras hasil a. pada satu sumbu pemodelan, prosedurnya sebagai berikut. Pertama membagi sumbu menjadi 3 segmen. Kedua mengisi segmen sumbu secara vertikal dengan benda dasar komponen tiang teras sehingga menghasilkan model tiang teras yang tergabung kontinu dan bervariasi.

## 5.2 Saran

Pada skripsi ini telah diperkenalkan prosedur modelisasi komponen penyusun tiang teras dan perangkaian komponen tiang teras pada satu sumbu pemodelan serta contoh perangkaian tiang teras rumah untuk menghasilkan bentuk tiang teras yang utuh. Diharapkan untuk penelitian selanjutnya metode ini dapat dikembangkan lagi menggunakan benda geometri ruang lainnya seperti prisma segi-$n$ dengan $n > 6$, kerucut, dan limas. Selain itu dapat ditawarkan penambahan relief yang lebih bervpariasi untuk sisi luar benda dasar komponen tiang teras.

# DAFTAR PUSTAKA


Anton,H. 1988. *Elementery Linear Algebra With Applications*. Terjemahan oleh P.Silaban dan I Nyoman Susila. Jakarta: Erlangga.

Astuti, P. 2014. Desain Rak Penataan Barang dengan Kurva dan Permukaan Tipe Natural, Hermit dan Bezier Kuadratik. Tesis, Jember : Jurusan Matematika Fakultas MIPA Universitas Jember.

Bastian, A. 2011. Desain Kap Lampu Duduk melalui Penggabungan Benda-benda Geometri Ruang. Tidak Diterbitkan. Skripsi. Jember : Jurusan Matematika Fakultas MIPA Universitas Jember.

Budhi, W Setya. 1995. *Aljabar Linier*. Jakarta:Gramedia.

Departemen Pendidikan Nasional. 2004. *Kamus Besar Bahasa Indonesia*. Jakarta: Balai Pustaka.

Fatkurotin. 2015. Konstruksi Botol Parfum melalui Penggabungan Benda Geometri Dasar Hasil Deformasi Prisma,Bola dan Tabung.Tidak Diterbitkan. 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 Elips.* Jember : Jurusan Matematika Fakultas MIPA Universitas Jember.

Kusno. 2009. *Geometri Rancang Bangun Studi tentang Desain dan Pemodelan Benda dengan Kurva dan Permukaan Berbantu Komputer*. Jember : Jurusan Matematika Fakultas MIPA Universitas Jember

Santoso,E. 2003. Modelisasi Bentuk Blok Aksesoris Bangunan yang Berbentuk Tiang Teras, Roster, dan Pot Bunga. Tidak Diterbitkan. Skripsi. Jember: Jurusan Matematika Fakultas MIPA Universitas Jember.

Suryadi, D. 1986. *Teori dan Soal Ilmu Analitik Ruang*. Jakarta: Ghalia Indonesia.

# LAMPIRAN

## Lampiran A. Modelisasi Komponen Penyusun Tiang Teras

### A.1 Deformasi Prisma Segienam Efek Puntiran

> t:=20, $\theta = 90°$
```
> theta:=Pi/2: #sudut putar#
> tpunt1:=2*t: tpunt2:=1/2*t: #tinggi#
> rpunt:=25: #jarak tipus ke titik sudut#

for i from 0 to 5 do
> cp[2*i+1]:="Red": cp[2*i+2]:="Pink":
a1[i+1]:=plot3d([(1-v)*((1-
    u)*rpunt*cos(Pi/3*i)+u*rpunt*cos(Pi/3*i+theta))+v*((1-
    u)*rpunt*cos(Pi/3*(i+1))+u*rpunt*cos(Pi/3*(i+1)+theta)),(1-
    v)*((1-u)*rpunt*sin(Pi/3*i)+u*rpunt*sin(Pi/3*i+theta))+v*((1-
    u)*rpunt*sin(Pi/3*(i+1))+u*rpunt*sin(Pi/3*(i+1)+theta)),(1-
    u)*tpunt1+u*tpunt2],u=0..1,v=0..1,color=cp[i+1]):
end do:
> puntiran:=display({a1[1],a1[2],a1[3],a1[4],a1[5],a1[6]}):
```

### A.2 Deformasi Prisma Segienam dengan Memberikan Lengkung pada Tiap sisinya

1. Lengkung cembung

Nilai untuk lengkung cekung adalah $\frac{1}{4}a \leq x_q, y_q \leq \frac{1}{2}a$, dan Nilai untuk lengkung cembung adalah $k > 1$.

```
> tcemb1:=0: tcemb2:=10:tcemb3:=20: #ketinggian titik kontrol#
> kcb:=2: #faktor pengali dilatasi segi-6#
> rcemb1:=30: rcemb2:=40: #titik kontrol pd sb x&y#
for k from 0 to 5 do
> ccb[2*k+1]:="Red": ccb[2*k+2]:="Pink":
c1[k+1]:=plot3d([(1-v)*((1-u)^2*rcemb1+2*(1-
   u)*u*rcemb2+u^2*rcemb1)*cos(Pi/3*k)+v*((1-u)^2*rcemb1+2*(1-
   u)*u*rcemb2+u^2*rcemb1)*cos(Pi/3*(k+1)),(1-v)*((1-
   u)^2*rcemb1+2*(1-u)*u*rcemb2+u^2*rcemb1)*sin(Pi/3*k)+v*((1-
   u)^2*rcemb1+2*(1-u)*u*rcemb2+u^2*rcemb1)*sin(Pi/3*(k+1)),(1-
   u)^2*tcemb1+2*(1-
   u)*u*tcemb2+u^2*tcemb3],u=0..1,v=0..1,color=ccb[k+1]):
end do:
> cembung:=display({c1[1],c1[2],c1[3],c1[4],c1[5],c1[6]}):
```

2. Lengkung cekung

```
> tcemb1:=0: tcemb2:=10:tcemb3:=20: #ketinggian titik kontrol#
> rcemb1:=40: rcemb2:=20: #titik kontrol pd sb x&y#

for k from 0 to 5 do
```

```
> ccb[2*k+1]:="Red": ccb[2*k+2]:="Pink":
> c1[k+1]:=plot3d([(1-v)*((1-u)^2*rcemb1+2*(1-
  u)*u*rcemb2+u^2*rcemb1)*cos(Pi/3*k)+v*((1-u)^2*rcemb1+2*(1-
  u)*u*rcemb2+u^2*rcemb1)*cos(Pi/3*(k+1)),(1-v)*((1-
  u)^2*rcemb1+2*(1-u)*u*rcemb2+u^2*rcemb1)*sin(Pi/3*k)+v*((1-
  u)^2*rcemb1+2*(1-u)*u*rcemb2+u^2*rcemb1)*sin(Pi/3*(k+1)),(1-
  u)^2*tcemb1+2*(1-
  u)*u*tcemb2+u^2*tcemb3],u=0..1,v=0..1,color=ccb[k+1]):
end do:
> cekung:=display({c1[1],c1[2],c1[3],c1[4],c1[5],c1[6]}):
```

### A.3 Deformasi Prisma Segienam Potong Pojok Bagian Atas

```
>x1:=u*(0)+(1-u)*30: y1:=u*(-40)+(1-u)*(-20): z1:=u*80+(1-
   u)*(80):
> x2:=u*(0)+(1-u)*20: y2:=u*(-40)+(1-u)*(-40): z2:=u*80+(1-
   u)*(60):
>x3:=u*(20)+(1-u)*30: y3:=u*(-40)+(1-u)*(-20): z3:=u*60+(1-
   u)*(80):
>a1:=plot3d([v*x1+(1-v)*x2,v*y1+(1-v)*y2, v*z1+(1-v)*z2], u=0..1,
   v=0..1, scaling=constrained, labels=[x,y,z], color="Blue",
   style=patchnogrid, lightmodel=light4):
>a2:=plot3d([v*x1+(1-v)*x3,v*y1+(1-v)*y3, v*z1+(1-v)*z3],u=0..1,
   v=0..1, scaling=constrained, labels=[x,y,z], color="Pink",
   style=patchnogrid, lightmodel=li ght4):
> x4:=u*(30)+(1-u)*30: y4:=u*(-20)+(1-u)*20: z4:=u*80+(1-u)*(80):
> x5:=u*(30)+(1-u)*40: y5:=u*(-20)+(1-u)*0: z5:=u*80+(1-u)*(60):
> x6:=u*(40)+(1-u)*3: y6:=u*(0)+(1-u)*2: z6:=u*6+(1-u)*(8):
>a3:=plot3d([v*x4+(1-v)*x6,v*y4+(1-v)*y6, v*z4+(1-v)*z6],u=0..1,
   v=0..1,scaling=constrained, labels=[x,y,z], color="Pink",
   style=patchnogrid, lightmodel=light4):
>a4:=plot3d([v*x4+(1-v)*x5,v*y4+(1-v)*y5, v*z4+(1- v)*z5] ,u=0..1,
   v=0..1, scaling=constrained, labels=[x,y,z], color="Pink",
   style=patchnogrid, lightmodel=light4):
> x7:=u*(30)+(1-u)*0: y7:=u*(20)+(1-u)*40: z7:=u*80+(1-u)*(80):
> x8:=u*(30)+(1-u)*20: y8:=u*(20)+(1-u)*40: z8:=u*80+(1-u)*(60):
> x9:=u*(20)+(1-u)*0: y9:=u*40+(1-u)*40: z9:=u*60+(1-u)*(80):
>a5:=plot3d([v*x7+(1-v)*x8,v*y7+(1-v)*y8, v*z7+(1-v)*z8],u=0..1,
   v=0..1,scaling=constrained,labels=[x,y,z], color="Pink",
   style=patchnogrid, lightmodel=light4):
>x10:=u*0+(1-u)*(-30): y10:=u*(40)+(1-u)*(20): z10:=u*80+(1-
   u)*(80):
>x11:=u*(0)+(1-u)*(-20): y11:=u*(40)+(1-u)*40: z11:=u*80+(1-
   u)*(60):
>x12:=u*(-20)+(1-u)*(-30): y12:=u*(40)+(1-u)*20: z12:=u*60+(1-
   u)*(80):
>a6:=plot3d([v*x10+(1-v)*x11,v*y10+(1-v)*y11, v*z10+(1-
   v)*z11],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
   color="Pink", style=patchnogrid, lightmodel=light4):
> x13:=u*(-30)+(1-u)*(-30): y13:=u*(20)+(1-u)*(-20): z13:=u*80+(1-
   u)*(80):
>x14:=u*(-30)+(1-u)*(-40): y14:=u*(20)+(1-u)*0: z14:=u*80+(1-
   u)*(60):
```

```
> x15:=u*(-40)+(1-u)*(-30): y15:=u*(0)+(1-u)*(-20): z15:=u*60+(1-
   u)*(80):
>a7:=plot3d([v*x13+(1-v)*x14,v*y13+(1-v)*y14, v*z13+(1-
   v)*z14],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
   color="Pink", style=patchnogrid, lightmodel=light4):
> x16:=u*(-30)+(1-u)*0: y16:=u*(-20)+(1-u)*(-40): z16:=u*80+(1-
   u)*(80):
> x17:=u*(-30)+(1-u)*(-20): y17:=u*(-20)+(1-u)*(-40):
   z17:=u*80+(1-u)*(60):
> x18:=u*(-20)+(1-u)*0: y18:=u*(-40)+(1-u)*(-40): z18:=u*60+(1-
   u)*(80):
> a8:=plot3d([v*x16+(1-v)*x17,v*y16+(1-v)*y17, v*z16+(1-
   v)*z17],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
   color="Pink", style=patchnogrid, lightmodel=light4):
> x19:=u*(-20)+(1-u)*(20): y19:=u*(-40)+(1-u)*(-40): z19:=u*0+(1-
   u)*(0):
> x20:=u*(20)+(1-u)*(40): y20:=u*(-40)+(1-u)*0: z20:=u*0+(1-
   u)*(0):
> x21:=u*(40)+(1-u)*(20): y21:=u*(0)+(1-u)*40: z21:=u*0+(1-u)*(0):
> x22:=u*(20)+(1-u)*(-20): y22:=u*(40)+(1-u)*40: z22:=u*0+(1-
   u)*(0):
> x23:=u*(-20)+(1-u)*(-40): y23:=u*(40)+(1-u)*0: z23:=u*0+(1-
   u)*(0):
> x24:=u*(-40)+(1-u)*(-20): y24:=u*(0)+(1-u)*(-40): z24:=u*0+(1-
   u)*(0):
> x25:=u*20+(1-u)*20: y25:=u*(-40)+(1-u)*(-40): z25:=u*0+(1-
   u)*(60):
> x26:=u*40+(1-u)*40: y26:=u*0+(1-u)*0: z26:=u*0+(1-u)*(60):
> x27:=u*20+(1-u)*20: y27:=u*40+(1-u)*40: z27:=u*0+(1-u)*(60):
> x28:=u*(-20)+(1-u)*(-20): y28:=u*40+(1-u)*40: z28:=u*0+(1-
   u)*(60):
> x29:=u*(-40)+(1-u)*(-40): y29:=u*0+(1-u)*0: z29:=u*0+(1-u)*(60):
> x30:=u*(-20)+(1-u)*(-20): y30:=u*(-40)+(1-u)*(-40): z30:=u*0+(1-
   u)*(60):
> a9:=plot3d([v*x19+(1-v)*x18,v*y19+(1-v)*y18, v*z19+(1-
   v)*z18],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
   color="Pink", style=patchnogrid, lightmodel=light4):
> a10:=plot3d([v*x19+(1-v)*x2,v*y19+(1-v)*y2, v*z19+(1-
   v)*z2],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
   color="Pink", style=patchnogrid, lightmodel=light4):
> a11:=plot3d([v*x3+(1-v)*x20,v*y3+(1-v)*y20, v*z3+(1-
   v)*z20],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
   color="Pink", style=patchnogrid, lightmodel=light4):
> a12:=plot3d([v*x5+(1-v)*x20,v*y5+(1-v)*y20, v*z5+(1-
   v)*z20],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
   color="Pink", style=patchnogrid, lightmodel=light4):
> a13:=plot3d([v*x6+(1-v)*x21,v*y6+(1-v)*y21, v*z6+(1-
   v)*z21],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
   color="Pink", style=patchnogrid, lightmodel=light4):
> a14:=plot3d([v*x8+(1-v)*x21,v*y8+(1-v)*y21, v*z8+(1-
   v)*z21],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
   color="Pink", style=patchnogrid, lightmodel=light4):
```

```
> a15:=plot3d([v*x9+(1-v)*x22,v*y9+(1-v)*y22, v*z9+(1-
    v)*z22],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
    color="Pink", style=patchnogrid, lightmodel=light4):
> a16:=plot3d([v*x11+(1-v)*x22,v*y11+(1-v)*y22, v*z11+(1-
    v)*z22],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
    color="Pink", style=patchnogrid, lightmodel=light4):
> a17:=plot3d([v*x12+(1-v)*x23,v*y12+(1-v)*y23, v*z12+(1-
    v)*z23],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
    color="Pink", style=patchnogrid, lightmodel=light4):
> a18:=plot3d([v*x14+(1-v)*x23,v*y14+(1-v)*y23, v*z14+(1-
    v)*z23],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
    color="Pink", style=patchnogrid, lightmodel=light4):
> a19:=plot3d([v*x15+(1-v)*x24,v*y15+(1-v)*y24, v*z15+(1-
    v)*z24],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
    color="Pink", style=patchnogrid, lightmodel=light4):
> a20:=plot3d([v*x17+(1-v)*x24,v*y17+(1-v)*y24, v*z17+(1-
    v)*z24],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
    color="Pink", style=patchnogrid, lightmodel=light4):
>display(a2,a3,a5,a6,a7,a8,a9,a10,a11,a12,a13,a14,a15,a16,a17,a18,
    a19,a20,axes=normal);
```

### A.4 Deformasi Tabung Modifikasi Selimut

$Q > r$

```
>c1:=plot3d([(30*(1-t)^2+40*2*(1-t)*t+t^2*30)*cos(v),(30*(1-
   t)^2+40*2*(1-t)*t+t^2*30)*sin(v),(0*(1-t)^2+5*2*(1-
   t)*t+t^2*10)],t=0..1,v=0..2*Pi):
>c2:=plot3d([(30*(1-t)^2+20*2*(1-t)*t+t^2*30)*cos(v),(30*(1-
   t)^2+20*2*(1-t)*t+t^2*30)*sin(v),(10*(1-t)^2+15*2*(1-
   t)*t+t^2*20)],t=0..1,v=0..2*Pi):
>c3:=plot3d([(30*(1-t)^2+40*2*(1-t)*t+t^2*30)*cos(v),(30*(1-
   t)^2+40*2*(1-t)*t+t^2*30)*sin(v),(20*(1-t)^2+40*2*(1-
   t)*t+t^2*50)],t=0..1,v=0..2*Pi):
>c4:=plot3d([(30*(1-t)^2+20*2*(1-t)*t+t^2*30)*cos(v),(30*(1-
   t)^2+20*2*(1-t)*t+t^2*30)*sin(v),(50*(1-t)^2+55*2*(1-
   t)*t+t^2*60)],t=0..1,v=0..2*Pi):
>display(c1,c2,c3,c4,axes=frame,style=patchnogrid,
   lightmodel=light4,scaling=constrained,color=pink);
```

$Q < r$

```
>d1:=plot3d([(30*(1-t)^2+40*2*(1-t)*t+t^2*30)*cos(v),(30*(1-
   t)^2+40*2*(1-t)*t+t^2*30)*sin(v),(0*(1-t)^2+5*2*(1-
   t)*t+t^2*10)],t=0..1,v=0..2*Pi):
>d2:=plot3d([(30*(1-t)^2+20*2*(1-t)*t+t^2*30)*cos(v),(30*(1-
   t)^2+20*2*(1-t)*t+t^2*30)*sin(v),(10*(1-t)^2+15*2*(1-
   t)*t+t^2*20)],t=0..1,v=0..2*Pi):
>d3:=plot3d([(30*(1-t)^2+20*2*(1-t)*t+t^2*30)*cos(v),(30*(1-
   t)^2+20*2*(1-t)*t+t^2*30)*sin(v),(20*(1-t)^2+40*2*(1-
   t)*t+t^2*50)],t=0..1,v=0..2*Pi):
>d4:=plot3d([(30*(1-t)^2+30*2*(1-t)*t+t^2*30)*cos(v),(30*(1-
   t)^2+30*2*(1-t)*t+t^2*30)*sin(v),(50*(1-t)^2+55*2*(1-
   t)*t+t^2*60)],t=0..1,v=0..2*Pi):
>display(d1,d2,d3,d4,axes=frame,style=patchnogrid,
   lightmodel=light4,scaling=constrained,color=pink);
```

### A.5 Deformasi Tabung dengan Memberi Pola pada Tutup

1. Pola busur gajil/ genap

```
>fa1:=plot3d([10*cos(u),10*sin(u),v],u=0..Pi/4,v=0..10,scaling=constrained, labels=[x,y,z], color="Pink", style=patchnogrid, lightmodel=light4):
>fa2:=rotate(fa1,0,0,Pi/2):fa21:=rotate(fa1,0,0,-Pi/2):fa22:=rotate(fa1,0,0,Pi):
>fb1:=plot3d([5*cos(u),5*sin(u),v],u=0..Pi/4,v=0..10,scaling=constrained, labels=[x,y,z], color="Pink", style=patchnogrid, lightmodel=light4):
>fb2:=rotate(fb1,0,0,Pi/2):fb21:=rotate(fb1,0,0,-Pi/2):fb22:=rotate(fb1,0,0,Pi):
>fd:=plot3d([5*cos(u),5*sin(u),v],u=0..2*Pi,v=0..10,scaling=constrained, labels=[x,y,z], color="Pink", style=patchnogrid, lightmodel=light4):
> x9:=u*(-10) +(1-u)*(-5):       y9:=u*0 +(1-u)*0: z9:=u*0 +(1-u)*(0):
> x10:=u*(-10) +(1-u)*(-5): y10:=u*0 +(1-u)*0: z10:=u*10 +(1-u)*(10):
>fc1:=plot3d([v*x9+(1-v)*x10,v*y9+(1-v)*y10, v*z9+(1-v)*z10],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z], color="Pink", style=patchnogrid, lightmodel=light4):
> fc2:=rotate(fc1,0,0,Pi/2):
> fc3:=rotate(fc1,0,0,Pi/4):
> fc4:=rotate(fc1,0,0,2*Pi/2.66):
> fc5:=rotate(fc1,0,0,Pi):
> fc6:=rotate(fc1,0,0,5*Pi/4):
> fc7:=rotate(fc1,0,0,3*Pi/2):
> fc8:=rotate(fc1,0,0,7*Pi/4):
>display(fa1,fa2,fa21,fa22,fb1,fb2,fb21,fb22,fc1,fc2,fc3,fc4,fc5,fc6,fc7,fc8,fd,axes=normal);
```

2. Pola 2 busur bersebelahan yang bertolak belakang

```
>a:=plot3d([25*cos(u),25*sin(u),v],u=0..Pi/2,v=-10..10,scaling=constrained, labels=[x,y,z], color="Pink", style=patchnogrid, lightmodel=light4):
>b:=plot3d([25*cos(u),25*sin(u),v],u=0..Pi/2,v=-10..10,scaling=constrained, labels=[x,y,z], color="Pink", style=patchnogrid, lightmodel=light4):
> b1:=rotate(a,0,0,Pi):
>a2:=plot3d([20*cos(u),20*sin(u),v],u=0..2*Pi,v=-10..10,scaling=constrained, labels=[x,y,z], color="Pink", style=patchnogrid, lightmodel=light4):
> x1:=u*(-20) +(1-u)*(-25): y1:=u*0 +(1-u)*0: z1:=u*(-10) +(1-u)*(-10):
```

```
> x2:=u*(-20) +(1-u)*(-25): y2:=u*0 +(1-u)*0: z2:=u*10 +(1-
  u)*(10):
>s1:=plot3d([v*x1+(1-v)*x2,v*y1+(1-v)*y2, v*z1+(1-v)*z2],  u=0..1,
  v=0..1,scaling=constrained,labels=[x,y,z], color="Pink",
  style=patchnogrid, lightmodel=light4):
> x3:=u*(25) +(1-u)*(20): y3:=u*0 +(1-u)*0: z3:=u*(-10) +(1-u)*(-
  10):
> x4:=u*(25) +(1-u)*(20): y4:=u*0 +(1-u)*0: z4:=u*10 +(1-u)*(10):
> s2:=plot3d([v*x3+(1-v)*x4,v*y3+(1-v)*y4, v*z3+(1-
  v)*z4],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> x5:=u*(0) +(1-u)*(0): y5:=u*25 +(1-u)*20: z5:=u*(-10) +(1-u)*(-
  10):
> x6:=u*(0) +(1-u)*(0): y6:=u*25+(1-u)*20: z6:=u*10 +(1-u)*(10):
> s3:=plot3d([v*x5+(1-v)*x6,v*y5+(1-v)*y6, v*z5+(1-
  v)*z6],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> x7:=u*(0) +(1-u)*(0): y7:=u*(-25) +(1-u)*(-20): z7:=u*(-10) +(1-
  u)*(-10):
> x8:=u*(0) +(1-u)*(0): y8:=u*(-25)+(1-u)*(-20): z8:=u*10 +(1-
  u)*(10):
> s4:=plot3d([v*x7+(1-v)*x8,v*y7+(1-v)*y8, v*z7+(1-
  v)*z8],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> display(b,b1,a2,s1,s2,s3,s4);
```

### A.6 Deformasi Pemotongan Bola

1. Pemotongan 1 bidang

```
>a1:=plot3d([20*sin(v)*cos(u),20*sin(v)*sin(u),(20*cos(v))],u=0..2
  *Pi,v=10..3/4*Pi,scaling=constrained, labels=[x,y,z],
  color="Violet", style=patchnogrid, lightmodel=light4):
> display(a1,axes=normal);
```

2. Pemotongan 2 bidang

```
>a2:=plot3d([20*sin(v)*cos(u),20*sin(v)*sin(u),(20*cos(v))],u=0..2
  *Pi,v=Pi/3..3*Pi/4,scaling=constrained, labels=[x,y,z],
  color="Violet", style=patchnogrid, lightmodel=light4):
> display(a2,axes=normal);
```

### A.7 Deformasi Bola dengan Dilatasi Jari-jari

```
> a1:=plot3d([3*sin(v)*cos(u),3*sin(v)*sin(u),(-
  3*cos(v))],u=0..2*Pi,v=(3/9)*Pi..(2/9)*Pi,scaling=constrained,
  labels=[x,y,z], color="Violet", style=patchnogrid,
  lightmodel=light4):
> a2:=plot3d([4*sin(v)*cos(u),4*sin(v)*sin(u),(-
  3*cos(v))],u=0..2*Pi,v=(4/9)*Pi..(3/9)*Pi,scaling=constrained,
  labels=[x,y,z], color="Violet", style=patchnogrid,
  lightmodel=light4):
> a3:=plot3d([3*sin(v)*cos(u),3*sin(v)*sin(u),(-
  3*cos(v))],u=0..2*Pi,v=(5/9)*Pi..(4/9)*Pi,scaling=constrained,
```

```
  labels=[x,y,z], color="Violet", style=patchnogrid,
  lightmodel=light4):
> a4:=plot3d([4*sin(v)*cos(u),4*sin(v)*sin(u),(-
  3*cos(v))],u=0..2*Pi,v=(6/9)*Pi..(5/9)*Pi,scaling=constrained,
  labels=[x,y,z], color="Violet", style=patchnogrid,
  lightmodel=light4):
> a5:=plot3d([3*sin(v)*cos(u),3*sin(v)*sin(u),(-
  3*cos(v))],u=0..2*Pi,v=(7/9)*Pi..(6/9)*Pi,scaling=constrained,
  labels=[x,y,z], color="Violet", style=patchnogrid,
  lightmodel=light4):
> a6:=plot3d([(1*(1-t)^2+1.5*2*(1-t)*t+t^2*2)*cos(v),(1*(1-
  t)^2+1.5*2*(1-t)*t+t^2*2)*sin(v),(-2.3*(1-t)^2+(-2.3)*2*(1-
  t)*t+t^2*(-2.3))],t=0..1,v=0..2*Pi,scaling=constrained,
  labels=[x,y,z], color="Violet", style=patchnogrid,
  lightmodel=light4):
> a7:=plot3d([(1*(1-t)^2+1.5*2*(1-t)*t+t^2*2)*cos(v),(1*(1-
  t)^2+1.5*2*(1-t)*t+t^2*2)*sin(v),(2.3*(1-t)^2+(2.3)*2*(1-
  t)*t+t^2*(2.3))],t=0..1,v=0..2*Pi,scaling=constrained,
  labels=[x,y,z], color="Violet", style=patchnogrid,
  lightmodel=light4):
> a8:=plot3d([(2.5*(1-t)^2+3.5*2*(1-t)*t+t^2*3.5)*cos(v),(2.5*(1-
  t)^2+3.5*2*(1-t)*t+t^2*3.5)*sin(v),(-1.5*(1-t)^2+(-1.5)*2*(1-
  t)*t+t^2*(-1.5))],t=0..1,v=0..2*Pi,scaling=constrained,
  labels=[x,y,z], color="Violet", style=patchnogrid,
  lightmodel=light4):
> a9:=plot3d([(2.5*(1-t)^2+3.5*2*(1-t)*t+t^2*3.5)*cos(v),(2.5*(1-
  t)^2+3.5*2*(1-t)*t+t^2*3.5)*sin(v),(1.5*(1-t)^2+(1.5)*2*(1-
  t)*t+t^2*(1.5))],t=0..1,v=0..2*Pi,scaling=constrained,
  labels=[x,y,z], color="Violet", style=patchnogrid,
  lightmodel=light4):
> a10:=plot3d([(2.5*(1-t)^2+3.5*2*(1-t)*t+t^2*4)*cos(v),(2.5*(1-
  t)^2+3.5*2*(1-t)*t+t^2*4)*sin(v),(-0.5*(1-t)^2+(-0.5)*2*(1-
  t)*t+t^2*(-0.5))],t=0..1,v=0..2*Pi,scaling=constrained,
  labels=[x,y,z], color="Violet", style=patchnogrid,
  lightmodel=light4):
> a11:=plot3d([(2.5*(1-t)^2+3.5*2*(1-t)*t+t^2*4)*cos(v),(2.5*(1-
  t)^2+3.5*2*(1-t)*t+t^2*4)*sin(v),(0.5*(1-t)^2+(0.5)*2*(1-
  t)*t+t^2*(0.5))],t=0..1,v=0..2*Pi,scaling=constrained,
  labels=[x,y,z], color="Violet", style=patchnogrid,
  lightmodel=light4):
> display(a1,a2,a3,a4,a5,a6,a7,a8,a9,a10,a11,axes=normal);
```

### A.8 Deformasi Bola dengan Efek Keratan

```
>a4:=plot3d([2*sin(v)*cos(u),2*sin(v)*sin(u),(2*cos(v))],u=20..2*P
  i,v=Pi/3..3*Pi/4,scaling=constrained, labels=[x,y,z],
  color="Violet", style=patchnogrid, lightmodel=light4):
> display(a4,axes=normal);
```

**Lampiran B. Perangkaian Tiang pada Sumbu Pemodelan**

- **Model 1 (1 benda dasar bagian alas, 1 benda  dasar bagian tengah, dan 1 benda dasar bagian atas)**

```
>b1:=plot3d([30*sin(v)*cos(u),30*sin(v)*sin(u),(30*cos(v))],u=20..
   2*Pi,v=Pi/3..3*Pi/4,scaling=constrained, labels=[x,y,z],
   color="Violet", style=patchnogrid, lightmodel=light4):
>b2:=plot3d([(26*(1-t)^2+20*2*(1-t)*t+t^2*15)*cos(v),(26*(1-
   t)^2+20*2*(1-t)*t+t^2*15)*sin(v),(13*(1-t)^2+13*2*(1-
   t)*t+t^2*13)],t=0..1,v=0..2*Pi,scaling=constrained,
   labels=[x,y,z], color="Violet", style=patchnogrid,
   lightmodel=light4):
>b3:=plot3d([(26*(1-t)^2+24*2*(1-t)*t+t^2*26)*cos(v),(26*(1-
   t)^2+24*2*(1-t)*t+t^2*26)*sin(v),(13*(1-t)^2+18*2*(1-
   t)*t+t^2*25)],t=0..1,v=0..2*Pi,scaling=constrained,
   labels=[x,y,z], color="Violet", style=patchnogrid,
   lightmodel=light4):
>b4:=plot3d([(26*(1-t)^2+15*2*(1-t)*t+t^2*26)*cos(v),(26*(1-
   t)^2+15*2*(1-t)*t+t^2*26)*sin(v),(25*(1-t)^2+30*2*(1-
   t)*t+t^2*35)],t=0..1,v=0..2*Pi,scaling=constrained,
   labels=[x,y,z], color="Violet", style=patchnogrid,
   lightmodel=light4):
>b5:=plot3d([(26*(1-t)^2+30*2*(1-t)*t+t^2*26)*cos(v),(26*(1-
   t)^2+30*2*(1-t)*t+t^2*26)*sin(v),(35*(1-t)^2+40*2*(1-
   t)*t+t^2*45)],t=0..1,v=0..2*Pi,scaling=constrained,
   labels=[x,y,z], color="Violet", style=patchnogrid,
   lightmodel=light4):
>b6:=plot3d([(26*(1-t)^2+15*2*(1-t)*t+t^2*10)*cos(v),(26*(1-
   t)^2+15*2*(1-t)*t+t^2*10)*sin(v),(45*(1-t)^2+45*2*(1-
   t)*t+t^2*45)],t=0..1,v=0..2*Pi,scaling=constrained,
   labels=[x,y,z], color="Violet", style=patchnogrid,
   lightmodel=light4):
> tcemb1:=0: tcemb2:=10:tcemb3:=30: #ketinggian titik kontrol#
kcb:=2: #faktor pengali dilatasi segi-6#
rcemb1:=18: rcemb2:=24: #titik kontrol pd sb x&y#

for k from 0 to 5 do
ccb[2*k+1]:="Violet": ccb[2*k+2]:="Violet":
c1[k+1]:=plot3d([(1-v)*((1-u)^2*rcemb1+2*(1-
   u)*u*rcemb2+u^2*rcemb1)*cos(Pi/3*k)+v*((1-u)^2*rcemb1+2*(1-
   u)*u*rcemb2+u^2*rcemb1)*cos(Pi/3*(k+1)),(1-v)*((1-
   u)^2*rcemb1+2*(1-u)*u*rcemb2+u^2*rcemb1)*sin(Pi/3*k)+v*((1-
   u)^2*rcemb1+2*(1-u)*u*rcemb2+u^2*rcemb1)*sin(Pi/3*(k+1)),(1-
   u)^2*tcemb1+2*(1-
   u)*u*tcemb2+u^2*tcemb3+45],u=0..1,v=0..1,color=ccb[k+1]):
end do:
b7:=display({c1[1],c1[2],c1[3],c1[4],c1[5],c1[6]},scaling=constrai
   ned, labels=[x,y,z], color="Violet", style=patchnogrid,
   lightmodel=light4):
> theta:=Pi/3: #sudut putar#
tpunt1:=80: tpunt2:=1/2*tpunt1: #tinggi#
rpunt:=18: #jarak tipus ke titik sudut#

for i from 0 to 5 do
```

```
cp[2*i+1]:="Violet": cp[2*i+2]:="Violet":
a1[i+1]:=plot3d([(1-v)*((1-
   u)*rpunt*cos(Pi/3*i)+u*rpunt*cos(Pi/3*i+theta))+v*((1-
   u)*rpunt*cos(Pi/3*(i+1))+u*rpunt*cos(Pi/3*(i+1)+theta)),(1-
   v)*((1-u)*rpunt*sin(Pi/3*i)+u*rpunt*sin(Pi/3*i+theta))+v*((1-
   u)*rpunt*sin(Pi/3*(i+1))+u*rpunt*sin(Pi/3*(i+1)+theta)),(1-
   u)*tpunt1+u*tpunt2+34],u=0..1,v=0..1,color=cp[i+1]):
end do:
b8:=display({a1[1],a1[2],a1[3],a1[4],a1[5],a1[6]},scaling=constrai
   ned, labels=[x,y,z], color="Violet", style=patchnogrid,
   lightmodel=light4):
> tcemb1:=0: tcemb2:=10:tcemb3:=30: #ketinggian titik kontrol#
kcb:=2: #faktor pengali dilatasi segi-6#
rcemb1:=18: rcemb2:=16: #titik kontrol pd sb x&y#

for k from 0 to 5 do
ccb[2*k+1]:="Violet": ccb[2*k+2]:="Violet":
c1[k+1]:=plot3d([(1-v)*((1-u)^2*rcemb1+2*(1-
   u)*u*rcemb2+u^2*rcemb1)*cos(Pi/3*k)+v*((1-u)^2*rcemb1+2*(1-
   u)*u*rcemb2+u^2*rcemb1)*cos(Pi/3*(k+1)),(1-v)*((1-
   u)^2*rcemb1+2*(1-u)*u*rcemb2+u^2*rcemb1)*sin(Pi/3*k)+v*((1-
   u)^2*rcemb1+2*(1-u)*u*rcemb2+u^2*rcemb1)*sin(Pi/3*(k+1)),(1-
   u)^2*tcemb1+2*(1-
   u)*u*tcemb2+u^2*tcemb3+114],u=0..1,v=0..1,color=ccb[k+1]):
end do:
b9:=display({c1[1],c1[2],c1[3],c1[4],c1[5],c1[6]},scaling=constrai
   ned, labels=[x,y,z], color="Violet", style=patchnogrid,
   lightmodel=light4):
> theta:=Pi/3: #sudut putar#
tpunt1:=80: tpunt2:=1/2*tpunt1: #tinggi#
rpunt:=18: #jarak tipus ke titik sudut#

for i from 0 to 5 do
cp[2*i+1]:="Violet": cp[2*i+2]:="Violet":
a1[i+1]:=plot3d([(1-v)*((1-
   u)*rpunt*cos(Pi/3*i)+u*rpunt*cos(Pi/3*i+theta))+v*((1-
   u)*rpunt*cos(Pi/3*(i+1))+u*rpunt*cos(Pi/3*(i+1)+theta)),(1-
   v)*((1-u)*rpunt*sin(Pi/3*i)+u*rpunt*sin(Pi/3*i+theta))+v*((1-
   u)*rpunt*sin(Pi/3*(i+1))+u*rpunt*sin(Pi/3*(i+1)+theta)),(1-
   u)*tpunt1+u*tpunt2+104],u=0..1,v=0..1,color=cp[i+1]):
end do:
b10:=display({a1[1],a1[2],a1[3],a1[4],a1[5],a1[6]},scaling=constra
   ined, labels=[x,y,z], color="Violet", style=patchnogrid,
   lightmodel=light4):
> tcemb1:=0: tcemb2:=10:tcemb3:=30: #ketinggian titik kontrol#
kcb:=2: #faktor pengali dilatasi segi-6#
rcemb1:=18: rcemb2:=16: #titik kontrol pd sb x&y#

for k from 0 to 5 do
ccb[2*k+1]:="Violet": ccb[2*k+2]:="Violet":
c1[k+1]:=plot3d([(1-v)*((1-u)^2*rcemb1+2*(1-
   u)*u*rcemb2+u^2*rcemb1)*cos(Pi/3*k)+v*((1-u)^2*rcemb1+2*(1-
   u)*u*rcemb2+u^2*rcemb1)*cos(Pi/3*(k+1)),(1-v)*((1-
   u)^2*rcemb1+2*(1-u)*u*rcemb2+u^2*rcemb1)*sin(Pi/3*k)+v*((1-
   u)^2*rcemb1+2*(1-u)*u*rcemb2+u^2*rcemb1)*sin(Pi/3*(k+1)),(1-
```

```
   u)^2*tcemb1+2*(1-
   u)*u*tcemb2+u^2*tcemb3+184],u=0..1,v=0..1,color=ccb[k+1]):
end do:
b11:=display({c1[1],c1[2],c1[3],c1[4],c1[5],c1[6]},scaling=constra
   ined, labels=[x,y,z], color="Violet", style=patchnogrid,
   lightmodel=light4):
> theta:=Pi/3: #sudut putar#
tpunt1:=80: tpunt2:=1/2*tpunt1: #tinggi#
rpunt:=18: #jarak tipus ke titik sudut#

for i from 0 to 5 do
cp[2*i+1]:="Violet": cp[2*i+2]:="Violet":
a1[i+1]:=plot3d([(1-v)*((1-
   u)*rpunt*cos(Pi/3*i)+u*rpunt*cos(Pi/3*i+theta))+v*((1-
   u)*rpunt*cos(Pi/3*(i+1))+u*rpunt*cos(Pi/3*(i+1)+theta)),(1-
   v)*((1-u)*rpunt*sin(Pi/3*i)+u*rpunt*sin(Pi/3*i+theta))+v*((1-
   u)*rpunt*sin(Pi/3*(i+1))+u*rpunt*sin(Pi/3*(i+1)+theta)),(1-
   u)*tpunt1+u*tpunt2+174],u=0..1,v=0..1,color=cp[i+1]):
end do:
b12:=display({a1[1],a1[2],a1[3],a1[4],a1[5],a1[6]},scaling=constra
   ined, labels=[x,y,z], color="Violet", style=patchnogrid,
   lightmodel=light4):
> tcemb1:=0: tcemb2:=20:tcemb3:=30: #ketinggian titik kontrol#
kcb:=2: #faktor pengali dilatasi segi-6#
rcemb1:=18: rcemb2:=24: #titik kontrol pd sb x&y#

for k from 0 to 5 do
ccb[2*k+1]:="Violet": ccb[2*k+2]:="Violet":
c1[k+1]:=plot3d([(1-v)*((1-u)^2*rcemb1+2*(1-
   u)*u*rcemb2+u^2*rcemb1)*cos(Pi/3*k)+v*((1-u)^2*rcemb1+2*(1-
   u)*u*rcemb2+u^2*rcemb1)*cos(Pi/3*(k+1)),(1-v)*((1-
   u)^2*rcemb1+2*(1-u)*u*rcemb2+u^2*rcemb1)*sin(Pi/3*k)+v*((1-
   u)^2*rcemb1+2*(1-u)*u*rcemb2+u^2*rcemb1)*sin(Pi/3*(k+1)),(1-
   u)^2*tcemb1+2*(1-
   u)*u*tcemb2+u^2*tcemb3+250],u=0..1,v=0..1,color=ccb[k+1]):
end do:
b13:=display({c1[1],c1[2],c1[3],c1[4],c1[5],c1[6]},scaling=constra
   ined, labels=[x,y,z], color="Violet", style=patchnogrid,
   lightmodel=light4):
>b14:=plot3d([(26*(1-t)^2+20*2*(1-t)*t+t^2*26)*cos(v),(26*(1-
   t)^2+20*2*(1-t)*t+t^2*26)*sin(v),(280*(1-t)^2+285*2*(1-
   t)*t+t^2*290)],t=0..1,v=0..2*Pi,scaling=constrained,
   labels=[x,y,z], color="Violet", style=patchnogrid,
   lightmodel=light4):
>b15:=plot3d([(26*(1-t)^2+28*2*(1-t)*t+t^2*15)*cos(v),(21*(1-
   t)^2+28*2*(1-t)*t+t^2*15)*sin(v),(280*(1-t)^2+280*2*(1-
   t)*t+t^2*280)],t=0..1,v=0..2*Pi,scaling=constrained,
   labels=[x,y,z], color="Violet", style=patchnogrid,
   lightmodel=light4):
>b16:=plot3d([(26*(1-t)^2+30*2*(1-t)*t+t^2*26)*cos(v),(26*(1-
   t)^2+30*2*(1-t)*t+t^2*26)*sin(v),(290*(1-t)^2+295*2*(1-
   t)*t+t^2*300)],t=0..1,v=0..2*Pi,scaling=constrained,
   labels=[x,y,z], color="Violet", style=patchnogrid,
   lightmodel=light4):
>b17:=plot3d([(26*(1-t)^2+20*2*(1-t)*t+t^2*26)*cos(v),(26*(1-
   t)^2+20*2*(1-t)*t+t^2*26)*sin(v),(300*(1-t)^2+305*2*(1-
```

```
   t)*t+t^2*310)],t=0..1,v=0..2*Pi,scaling=constrained,
   labels=[x,y,z], color="Violet", style=patchnogrid,
   lightmodel=light4):
>b18:=plot3d([(26*(1-t)^2+15*2*(1-t)*t+t^2*0)*cos(v),(26*(1-
   t)^2+15*2*(1-t)*t+t^2*0)*sin(v),(310*(1-t)^2+310*2*(1-
   t)*t+t^2*310)],t=0..1,v=0..2*Pi,scaling=constrained,
   labels=[x,y,z], color="Violet", style=patchnogrid,
   lightmodel=light4):
>display(b2,b3,b4,b5,b6,b7,b8,b9,b10,b11,b12,b13,b14,b15,b16,b17,b
   18,axes=normal);
```

- **Model 2 (1 benda dasar bagian alas, 1 benda dasar bagian tengah, dan 2 benda dasar bagian atas)**

```
> b1:=plot3d([(2.7*(1-t)^2+5*2*(1-t)*t+t^2*2.7)*cos(v),(2.7*(1-
   t)^2+5*2*(1-t)*t+t^2*2.7)*sin(v),((-2)*(1-t)^2+(-1)*2*(1-
   t)*t+t^2*0)],t=0..1,v=0..2*Pi,color=pink):
> b2:=plot3d([(2.7*(1-t)^2+4*2*(1-t)*t+t^2*2.7)*cos(v),(2.7*(1-
   t)^2+4*2*(1-t)*t+t^2*2.7)*sin(v),(0*(1-t)^2+1*2*(1-
   t)*t+t^2*2)],t=0..1,v=0..2*Pi,color=pink):
> b3:=plot3d([(2.7*(1-t)^2+1.7*2*(1-t)*t+t^2*2.7)*cos(v),(2.7*(1-
   t)^2+1.7*2*(1-t)*t+t^2*2.7)*sin(v),(2*(1-t)^2+4*2*(1-
   t)*t+t^2*6)],t=0..1,v=0..2*Pi,color=pink):
> b4:=plot3d([(2.7*(1-t)^2+4*2*(1-t)*t+t^2*2.7)*cos(v),(2.7*(1-
   t)^2+4*2*(1-t)*t+t^2*2.7)*sin(v),(6*(1-t)^2+7*2*(1-
   t)*t+t^2*8)],t=0..1,v=0..2*Pi,color=pink):
> b5:=plot3d([(2.7*(1-t)^2+2*2*(1-t)*t+t^2*1)*cos(v),(2.7*(1-
   t)^2+2*2*(1-t)*t+t^2*1)*sin(v),(8*(1-t)^2+8*2*(1-
   t)*t+t^2*8)],t=0..1,v=0..2*Pi,color=pink):
> theta:=Pi/3: #sudut putar#
tpunt1:=0: tpunt2:=4: #tinggi#
rpunt:=2: #jarak tipus ke titik sudut#

for i from 0 to 5 do
cp[2*i+1]:="Pink": cp[2*i+2]:="Pink":
d1[i+1]:=plot3d([(1-v)*((1-
   u)*rpunt*cos(Pi/3*i)+u*rpunt*cos(Pi/3*i+theta))+v*((1-
   u)*rpunt*cos(Pi/3*(i+1))+u*rpunt*cos(Pi/3*(i+1)+theta)),(1-
   v)*((1-u)*rpunt*sin(Pi/3*i)+u*rpunt*sin(Pi/3*i+theta))+v*((1-
   u)*rpunt*sin(Pi/3*(i+1))+u*rpunt*sin(Pi/3*(i+1)+theta)),(1-
   u)*tpunt1+u*tpunt2+8],u=0..1,v=0..1,color=cp[i+1]):
end do:
a1:=display({d1[1],d1[2],d1[3],d1[4],d1[5],d1[6]}):
> tcemb1:=0: tcemb2:=2:tcemb3:=4: #ketinggian titik kontrol#
kcb:=2: #faktor pengali dilatasi segi-6#
rcemb1:=2: rcemb2:=1: #titik kontrol pd sb x&y#

for k from 0 to 5 do
ccb[2*k+1]:="Pink": ccb[2*k+2]:="Pink":
c1[k+1]:=plot3d([(1-v)*((1-u)^2*rcemb1+2*(1-
   u)*u*rcemb2+u^2*rcemb1)*cos(Pi/3*k)+v*((1-u)^2*rcemb1+2*(1-
   u)*u*rcemb2+u^2*rcemb1)*cos(Pi/3*(k+1)),(1-v)*((1-
```

```
   u)^2*rcemb1+2*(1-u)*u*rcemb2+u^2*rcemb1)*sin(Pi/3*k)+v*((1-
   u)^2*rcemb1+2*(1-u)*u*rcemb2+u^2*rcemb1)*sin(Pi/3*(k+1)),(1-
   u)^2*tcemb1+2*(1-
   u)*u*tcemb2+u^2*tcemb3+12],u=0..1,v=0..1,color=ccb[k+1]):
end do:
a2:=display({c1[1],c1[2],c1[3],c1[4],c1[5],c1[6]}):
> tcemb1:=0: tcemb2:=2:tcemb3:=4: #ketinggian titik kontrol#
kcb:=2: #faktor pengali dilatasi segi-6#
rcemb1:=2: rcemb2:=1: #titik kontrol pd sb x&y#

for k from 0 to 5 do
ccb[2*k+1]:="Pink": ccb[2*k+2]:="Pink":
c1[k+1]:=plot3d([(1-v)*((1-u)^2*rcemb1+2*(1-
   u)*u*rcemb2+u^2*rcemb1)*cos(Pi/3*k)+v*((1-u)^2*rcemb1+2*(1-
   u)*u*rcemb2+u^2*rcemb1)*cos(Pi/3*(k+1)),(1-v)*((1-
   u)^2*rcemb1+2*(1-u)*u*rcemb2+u^2*rcemb1)*sin(Pi/3*k)+v*((1-
   u)^2*rcemb1+2*(1-u)*u*rcemb2+u^2*rcemb1)*sin(Pi/3*(k+1)),(1-
   u)^2*tcemb1+2*(1-
   u)*u*tcemb2+u^2*tcemb3+16],u=0..1,v=0..1,color=ccb[k+1]):
end do:
a3:=display({c1[1],c1[2],c1[3],c1[4],c1[5],c1[6]}):
> tcemb1:=0: tcemb2:=2:tcemb3:=4: #ketinggian titik kontrol#
kcb:=2: #faktor pengali dilatasi segi-6#
rcemb1:=2: rcemb2:=2.5: #titik kontrol pd sb x&y#

for k from 0 to 5 do
ccb[2*k+1]:="Pink": ccb[2*k+2]:="Pink":
c1[k+1]:=plot3d([(1-v)*((1-u)^2*rcemb1+2*(1-
   u)*u*rcemb2+u^2*rcemb1)*cos(Pi/3*k)+v*((1-u)^2*rcemb1+2*(1-
   u)*u*rcemb2+u^2*rcemb1)*cos(Pi/3*(k+1)),(1-v)*((1-
   u)^2*rcemb1+2*(1-u)*u*rcemb2+u^2*rcemb1)*sin(Pi/3*k)+v*((1-
   u)^2*rcemb1+2*(1-u)*u*rcemb2+u^2*rcemb1)*sin(Pi/3*(k+1)),(1-
   u)^2*tcemb1+2*(1-
   u)*u*tcemb2+u^2*tcemb3+20],u=0..1,v=0..1,color=ccb[k+1]):
end do:
a4:=display({c1[1],c1[2],c1[3],c1[4],c1[5],c1[6]}):
> tcemb1:=0: tcemb2:=2:tcemb3:=4: #ketinggian titik kontrol#
kcb:=2: #faktor pengali dilatasi segi-6#
rcemb1:=2: rcemb2:=1: #titik kontrol pd sb x&y#

for k from 0 to 5 do
ccb[2*k+1]:="Pink": ccb[2*k+2]:="Pink":
c1[k+1]:=plot3d([(1-v)*((1-u)^2*rcemb1+2*(1-
   u)*u*rcemb2+u^2*rcemb1)*cos(Pi/3*k)+v*((1-u)^2*rcemb1+2*(1-
   u)*u*rcemb2+u^2*rcemb1)*cos(Pi/3*(k+1)),(1-v)*((1-
   u)^2*rcemb1+2*(1-u)*u*rcemb2+u^2*rcemb1)*sin(Pi/3*k)+v*((1-
   u)^2*rcemb1+2*(1-u)*u*rcemb2+u^2*rcemb1)*sin(Pi/3*(k+1)),(1-
   u)^2*tcemb1+2*(1-
   u)*u*tcemb2+u^2*tcemb3+24],u=0..1,v=0..1,color=ccb[k+1]):
end do:
a5:=display({c1[1],c1[2],c1[3],c1[4],c1[5],c1[6]}):
> tcemb1:=0: tcemb2:=2:tcemb3:=4: #ketinggian titik kontrol#
kcb:=2: #faktor pengali dilatasi segi-6#
rcemb1:=2: rcemb2:=1: #titik kontrol pd sb x&y#

for k from 0 to 5 do
```

```
ccb[2*k+1]:="Pink": ccb[2*k+2]:="Pink":
c1[k+1]:=plot3d([(1-v)*((1-u)^2*rcemb1+2*(1-
   u)*u*rcemb2+u^2*rcemb1)*cos(Pi/3*k)+v*((1-u)^2*rcemb1+2*(1-
   u)*u*rcemb2+u^2*rcemb1)*cos(Pi/3*(k+1)),(1-v)*((1-
   u)^2*rcemb1+2*(1-u)*u*rcemb2+u^2*rcemb1)*sin(Pi/3*k)+v*((1-
   u)^2*rcemb1+2*(1-u)*u*rcemb2+u^2*rcemb1)*sin(Pi/3*(k+1)),(1-
   u)^2*tcemb1+2*(1-
   u)*u*tcemb2+u^2*tcemb3+28],u=0..1,v=0..1,color=ccb[k+1]):
end do:
a6:=display({c1[1],c1[2],c1[3],c1[4],c1[5],c1[6]}):
> theta:=Pi/3: #sudut putar#
tpunt1:=0: tpunt2:=4: #tinggi#
rpunt:=2: #jarak tipus ke titik sudut#

for i from 0 to 5 do
cp[2*i+1]:="Pink": cp[2*i+2]:="Pink":
d1[i+1]:=plot3d([(1-v)*((1-
   u)*rpunt*cos(Pi/3*i)+u*rpunt*cos(Pi/3*i+theta))+v*((1-
   u)*rpunt*cos(Pi/3*(i+1))+u*rpunt*cos(Pi/3*(i+1)+theta)),(1-
   v)*((1-u)*rpunt*sin(Pi/3*i)+u*rpunt*sin(Pi/3*i+theta))+v*((1-
   u)*rpunt*sin(Pi/3*(i+1))+u*rpunt*sin(Pi/3*(i+1)+theta)),(1-
   u)*tpunt1+u*tpunt2+32],u=0..1,v=0..1,color=cp[i+1]):
end do:
a7:=display({d1[1],d1[2],d1[3],d1[4],d1[5],d1[6]}):
>
   a8:=plot3d([2.8*sin(v)*cos(u),2.8*sin(v)*sin(u),(2*cos(v)+37.3)
   ],u=50..2*Pi,v=Pi/3..3*Pi/4,scaling=constrained,
   labels=[x,y,z], color="Violet", style=patchnogrid,
   lightmodel=light4):
> a9:=plot3d([(1*(1-t)^2+2*2*(1-t)*t+t^2*2.8)*cos(v),(1*(1-
   t)^2+2*2*(1-t)*t+t^2*2.8)*sin(v),(38.3*(1-t)^2+38.3*2*(1-
   t)*t+t^2*38.3)],t=0..1,v=0..2*Pi,scaling=constrained,
   labels=[x,y,z], color="Violet", style=patchnogrid,
   lightmodel=light4):
> a10:=plot3d([(2.8*(1-t)^2+4*2*(1-t)*t+t^2*2.8)*cos(v),(2.8*(1-
   t)^2+4*2*(1-t)*t+t^2*2.8)*sin(v),(38.3*(1-t)^2+39.3*2*(1-
   t)*t+t^2*40.5)],t=0..1,v=0..2*Pi,scaling=constrained,
   labels=[x,y,z], color="Pink", style=patchnogrid,
   lightmodel=light4):
> a11:=plot3d([(2.8*(1-t)^2+1.5*2*(1-t)*t+t^2*2.8)*cos(v),(2.8*(1-
   t)^2+1.5*2*(1-t)*t+t^2*2.8)*sin(v),(40.5*(1-t)^2+41.5*2*(1-
   t)*t+t^2*42.5)],t=0..1,v=0..2*Pi,scaling=constrained,
   labels=[x,y,z], color="Pink", style=patchnogrid,
   lightmodel=light4):
> a12:=plot3d([(2.8*(1-t)^2+2.8*2*(1-t)*t+t^2*2.8)*cos(v),(2.8*(1-
   t)^2+2.8*2*(1-t)*t+t^2*2.8)*sin(v),(42.5*(1-t)^2+42.5*2*(1-
   t)*t+t^2*42.5)],t=0..1,v=0..2*Pi,scaling=constrained,
   labels=[x,y,z], color="Pink", style=patchnogrid,
   lightmodel=light4):
>display(a1,a2,a3,a4,a5,a6,a7,a8,a9,a10,a11,a12,b1,b2,b3,b4,b5,axe
   s=normal,scaling=constrained,color=pink,lightmodel=light4,style
   =patchnogrid);
```

- **Model 3 (2 benda dasar bagian alas, 1 benda dasar bagian tengah, dan 1 benda dasar bagian atas)**

```
> a1:=plot3d([7*sin(v)*cos(u),7*sin(v)*sin(u),(4*cos(v)+(-
  4))],u=45..2*Pi,v=Pi/3..3*Pi/4,scaling=constrained,
  labels=[x,y,z], color="Violet", style=patchnogrid,
  lightmodel=light4):
> e4:=plot3d([(6*(1-t)^2+5*2*(1-t)*t+t^2*3)*cos(v),(6*(1-
  t)^2+5*2*(1-t)*t+t^2*3)*sin(v),(-4*(1-t)^2+(-4)*2*(1-
  t)*t+t^2*(-4))],t=0..1,v=0..2*Pi):
> e5:=plot3d([(6*(1-t)^2+7*2*(1-t)*t+t^2*6)*cos(v),(6*(1-
  t)^2+7*2*(1-t)*t+t^2*6)*sin(v),(-4*(1-t)^2+(-2)*2*(1-
  t)*t+t^2*(0))],t=0..1,v=0..2*Pi):
> e6:=plot3d([(6*(1-t)^2+3*2*(1-t)*t+t^2*6)*cos(v),(6*(1-
  t)^2+3*2*(1-t)*t+t^2*6)*sin(v),(0*(1-t)^2+2*2*(1-
  t)*t+t^2*4)],t=0..1,v=0..2*Pi):
> e7:=plot3d([(6*(1-t)^2+8*2*(1-t)*t+t^2*6)*cos(v),(6*(1-
  t)^2+8*2*(1-t)*t+t^2*6)*sin(v),(4*(1-t)^2+5*2*(1-
  t)*t+t^2*6)],t=0..1,v=0..2*Pi):
> e8:=plot3d([(6*(1-t)^2+3*2*(1-t)*t+t^2*1)*cos(v),(6*(1-
  t)^2+3*2*(1-t)*t+t^2*1)*sin(v),(6*(1-t)^2+6*2*(1-
  t)*t+t^2*6)],t=0..1,v=0..2*Pi):
> e9:=plot3d([(6*(1-t)^2+3*2*(1-t)*t+t^2*1)*cos(v),(6*(1-
  t)^2+3*2*(1-t)*t+t^2*1)*sin(v),(40*(1-t)^2+40*2*(1-
  t)*t+t^2*40)],t=0..1,v=0..2*Pi):
> e10:=plot3d([(6*(1-t)^2+3*2*(1-t)*t+t^2*6)*cos(v),(6*(1-
  t)^2+3*2*(1-t)*t+t^2*6)*sin(v),(40*(1-t)^2+42*2*(1-
  t)*t+t^2*44)],t=0..1,v=0..2*Pi):
> e11:=plot3d([(6*(1-t)^2+8*2*(1-t)*t+t^2*6)*cos(v),(6*(1-
  t)^2+8*2*(1-t)*t+t^2*6)*sin(v),(44*(1-t)^2+46*2*(1-
  t)*t+t^2*48)],t=0..1,v=0..2*Pi):
> e12:=plot3d([(6*(1-t)^2+3*2*(1-t)*t+t^2*6)*cos(v),(6*(1-
  t)^2+3*2*(1-t)*t+t^2*6)*sin(v),(48*(1-t)^2+50*2*(1-
  t)*t+t^2*51)],t=0..1,v=0..2*Pi):
> x1:=u*(0)+(1-u)*3: y1:=u*(-4)+(1-u)*(-2): z1:=u*20+(1-u)*(20):
> x2:=u*(0)+(1-u)*2: y2:=u*(-4)+(1-u)*(-4): z2:=u*20+(1-u)*(18):
> x3:=u*(2)+(1-u)*3: y3:=u*(-4)+(1-u)*(-2): z3:=u*18+(1-u)*(20):
> a1:=plot3d([v*x1+(1-v)*x3,v*y1+(1-v)*y3, v*z1+(1-
  v)*z3],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> x4:=u*(3)+(1-u)*3: y4:=u*(-2)+(1-u)*2: z4:=u*20+(1-u)*(20):
> x5:=u*(3)+(1-u)*4: y5:=u*(-2)+(1-u)*0: z5:=u*20+(1-u)*(18):
> x6:=u*(4)+(1-u)*3: y6:=u*(0)+(1-u)*2: z6:=u*18+(1-u)*(20):
> a2:=plot3d([v*x4+(1-v)*x6,v*y4+(1-v)*y6, v*z4+(1-
  v)*z6],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> x7:=u*(3)+(1-u)*0: y7:=u*(2)+(1-u)*4: z7:=u*20+(1-u)*(20):
> x8:=u*(3)+(1-u)*2: y8:=u*(2)+(1-u)*4: z8:=u*20+(1-u)*(18):
> x9:=u*(2)+(1-u)*0: y9:=u*4+(1-u)*4: z9:=u*18+(1-u)*(20):
> a3:=plot3d([v*x7+(1-v)*x8,v*y7+(1-v)*y8, v*z7+(1-
  v)*z8],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> x10:=u*0+(1-u)*(-3): y10:=u*(4)+(1-u)*(2): z10:=u*20+(1-u)*(20):
```

```
> x11:=u*(0)+(1-u)*(-2): y11:=u*(4)+(1-u)*4: z11:=u*20+(1-u)*(18):
> x12:=u*(-2)+(1-u)*(-3): y12:=u*(4)+(1-u)*2: z12:=u*18+(1-
  u)*(20):
> a4:=plot3d([v*x10+(1-v)*x11,v*y10+(1-v)*y11, v*z10+(1-
  v)*z11],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> x13:=u*(-3)+(1-u)*(-3): y13:=u*(2)+(1-u)*(-2): z13:=u*20+(1-
  u)*(20):
> x14:=u*(-3)+(1-u)*(-4): y14:=u*(2)+(1-u)*0: z14:=u*20+(1-
  u)*(18):
> x15:=u*(-4)+(1-u)*(-3): y15:=u*(0)+(1-u)*(-2): z15:=u*18+(1-
  u)*(20):
> a5:=plot3d([v*x13+(1-v)*x14,v*y13+(1-v)*y14, v*z13+(1-
  v)*z14],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> x16:=u*(-3)+(1-u)*0: y16:=u*(-2)+(1-u)*(-4): z16:=u*20+(1-
  u)*(20):
> x17:=u*(-3)+(1-u)*(-2): y17:=u*(-2)+(1-u)*(-4): z17:=u*20+(1-
  u)*(18):
> x18:=u*(-2)+(1-u)*0: y18:=u*(-4)+(1-u)*(-4): z18:=u*18+(1-
  u)*(20):
> a6:=plot3d([v*x16+(1-v)*x17,v*y16+(1-v)*y17, v*z16+(1-
  v)*z17],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> x31:=u*(0)+(1-u)*3: y31:=u*(-4)+(1-u)*(-2): z31:=u*6+(1-u)*(6):
> x32:=u*(0)+(1-u)*2: y32:=u*(-4)+(1-u)*(-4): z32:=u*6+(1-u)*(8):
> x33:=u*(2)+(1-u)*3: y33:=u*(-4)+(1-u)*(-2): z33:=u*8+(1-u)*(6):
> b1:=plot3d([v*x31+(1-v)*x32,v*y31+(1-v)*y32, v*z31+(1-
  v)*z32],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> x34:=u*(3)+(1-u)*3: y34:=u*(-2)+(1-u)*2: z34:=u*6+(1-u)*(6):
> x35:=u*(3)+(1-u)*4: y35:=u*(-2)+(1-u)*0: z35:=u*6+(1-u)*(8):
> x36:=u*(4)+(1-u)*3: y36:=u*(0)+(1-u)*2: z36:=u*8+(1-u)*(6):
> b2:=plot3d([v*x34+(1-v)*x36,v*y34+(1-v)*y36, v*z34+(1-
  v)*z36],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> x37:=u*(3)+(1-u)*0: y37:=u*(2)+(1-u)*4: z37:=u*6+(1-u)*(6):
> x38:=u*(3)+(1-u)*2: y38:=u*(2)+(1-u)*4: z38:=u*6+(1-u)*(8):
> x39:=u*(2)+(1-u)*0: y39:=u*4+(1-u)*4: z39:=u*8+(1-u)*(6):
> b3:=plot3d([v*x37+(1-v)*x38,v*y37+(1-v)*y38, v*z37+(1-
  v)*z38],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> x40:=u*0+(1-u)*(-3): y40:=u*(4)+(1-u)*(2): z40:=u*6+(1-u)*(6):
> x41:=u*(0)+(1-u)*(-2): y41:=u*(4)+(1-u)*4: z41:=u*6+(1-u)*(8):
> x42:=u*(-2)+(1-u)*(-3): y42:=u*(4)+(1-u)*2: z42:=u*8+(1-u)*(6):
> b4:=plot3d([v*x40+(1-v)*x41,v*y40+(1-v)*y41, v*z40+(1-
  v)*z41],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> x43:=u*(-3)+(1-u)*(-3): y43:=u*(2)+(1-u)*(-2): z43:=u*6+(1-
  u)*(6):
> x44:=u*(-3)+(1-u)*(-4): y44:=u*(2)+(1-u)*0: z44:=u*6+(1-u)*(8):
> x45:=u*(-4)+(1-u)*(-3): y45:=u*(0)+(1-u)*(-2): z45:=u*8+(1-
  u)*(6):
```

```
> b5:=plot3d([v*x43+(1-v)*x44,v*y43+(1-v)*y44, v*z43+(1-
  v)*z44],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> x46:=u*(-3)+(1-u)*0: y46:=u*(-2)+(1-u)*(-4): z46:=u*6+(1-u)*(6):
> x47:=u*(-3)+(1-u)*(-2): y47:=u*(-2)+(1-u)*(-4): z47:=u*6+(1-
  u)*(8):
> x48:=u*(-2)+(1-u)*0: y48:=u*(-4)+(1-u)*(-4): z48:=u*8+(1-u)*(6):
> b6:=plot3d([v*x46+(1-v)*x47,v*y46+(1-v)*y47, v*z46+(1-
  v)*z47],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> s1:=plot3d([v*x2+(1-v)*x32,v*y2+(1-v)*y32, v*z2+(1-
  v)*z32],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> s2:=plot3d([v*x3+(1-v)*x33,v*y3+(1-v)*y33, v*z3+(1-
  v)*z33],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> s3:=plot3d([v*x5+(1-v)*x35,v*y5+(1-v)*y35, v*z5+(1-
  v)*z35],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> s4:=plot3d([v*x6+(1-v)*x36,v*y6+(1-v)*y36, v*z3+(1-
  v)*z36],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> s5:=plot3d([v*x8+(1-v)*x38,v*y8+(1-v)*y38, v*z8+(1-
  v)*z38],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> s6:=plot3d([v*x9+(1-v)*x39,v*y9+(1-v)*y39, v*z3+(1-
  v)*z39],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> s7:=plot3d([v*x11+(1-v)*x41,v*y11+(1-v)*y41, v*z11+(1-
  v)*z41],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> s8:=plot3d([v*x12+(1-v)*x42,v*y12+(1-v)*y42, v*z12+(1-
  v)*z42],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> s9:=plot3d([v*x14+(1-v)*x44,v*y14+(1-v)*y44, v*z14+(1-
  v)*z44],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> s10:=plot3d([v*x15+(1-v)*x45,v*y15+(1-v)*y45, v*z15+(1-
  v)*z45],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> s11:=plot3d([v*x17+(1-v)*x47,v*y17+(1-v)*y47, v*z17+(1-
  v)*z47],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> s12:=plot3d([v*x18+(1-v)*x48,v*y18+(1-v)*y48, v*z18+(1-
  v)*z48],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> x51:=u*(0)+(1-u)*3: y51:=u*(-4)+(1-u)*(-2): z51:=u*40+(1-
  u)*(40):
> x52:=u*(0)+(1-u)*2: y52:=u*(-4)+(1-u)*(-4): z52:=u*40+(1-
  u)*(38):
> x53:=u*(2)+(1-u)*3: y53:=u*(-4)+(1-u)*(-2): z53:=u*38+(1-
  u)*(40):
```

```
> c1:=plot3d([v*x51+(1-v)*x53,v*y51+(1-v)*y53, v*z51+(1-
  v)*z53],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> x54:=u*(3)+(1-u)*3: y54:=u*(-2)+(1-u)*2: z54:=u*40+(1-u)*(40):
> x55:=u*(3)+(1-u)*4: y55:=u*(-2)+(1-u)*0: z55:=u*40+(1-u)*(38):
> x56:=u*(4)+(1-u)*3: y56:=u*(0)+(1-u)*2: z56:=u*38+(1-u)*(40):
> c2:=plot3d([v*x54+(1-v)*x56,v*y54+(1-v)*y56, v*z54+(1-
  v)*z56],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> x57:=u*(3)+(1-u)*0: y57:=u*(2)+(1-u)*4: z57:=u*40+(1-u)*(40):
> x58:=u*(3)+(1-u)*2: y58:=u*(2)+(1-u)*4: z58:=u*40+(1-u)*(38):
> x59:=u*(2)+(1-u)*0: y59:=u*4+(1-u)*4: z59:=u*38+(1-u)*(40):
> c3:=plot3d([v*x57+(1-v)*x58,v*y57+(1-v)*y58, v*z57+(1-
  v)*z58],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> x60:=u*0+(1-u)*(-3): y60:=u*(4)+(1-u)*(2): z60:=u*40+(1-u)*(40):
> x61:=u*(0)+(1-u)*(-2): y61:=u*(4)+(1-u)*4: z61:=u*40+(1-u)*(38):
> x62:=u*(-2)+(1-u)*(-3): y62:=u*(4)+(1-u)*2: z62:=u*38+(1-
  u)*(40):
> c4:=plot3d([v*x60+(1-v)*x61,v*y60+(1-v)*y61, v*z60+(1-
  v)*z61],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> x63:=u*(-3)+(1-u)*(-3): y63:=u*(2)+(1-u)*(-2): z63:=u*40+(1-
  u)*(40):
> x64:=u*(-3)+(1-u)*(-4): y64:=u*(2)+(1-u)*0: z64:=u*40+(1-
  u)*(38):
> x65:=u*(-4)+(1-u)*(-3): y65:=u*(0)+(1-u)*(-2): z65:=u*38+(1-
  u)*(40):
> c5:=plot3d([v*x63+(1-v)*x64,v*y63+(1-v)*y64, v*z63+(1-
  v)*z64],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> x66:=u*(-3)+(1-u)*0: y66:=u*(-2)+(1-u)*(-4): z66:=u*40+(1-
  u)*(40):
> x67:=u*(-3)+(1-u)*(-2): y67:=u*(-2)+(1-u)*(-4): z67:=u*40+(1-
  u)*(38):
> x68:=u*(-2)+(1-u)*0: y68:=u*(-4)+(1-u)*(-4): z68:=u*38+(1-
  u)*(40):
> c6:=plot3d([v*x66+(1-v)*x67,v*y66+(1-v)*y67, v*z66+(1-
  v)*z67],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid,
  lightmodel=light4):x31:=u*(0)+(1-u)*3:
> x71:=u*(0)+(1-u)*3: y31:=u*(-4)+(1-u)*(-2): z31:=u*20+(1-
  u)*(20):
> x72:=u*(0)+(1-u)*2: y72:=u*(-4)+(1-u)*(-4): z72:=u*20+(1-
  u)*(22):
> x73:=u*(2)+(1-u)*3: y73:=u*(-4)+(1-u)*(-2): z73:=u*22+(1-
  u)*(20):
> d1:=plot3d([v*x71+(1-v)*x72,v*y71+(1-v)*y72, v*z71+(1-
  v)*z72],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> x74:=u*(3)+(1-u)*3: y74:=u*(-2)+(1-u)*2: z74:=u*20+(1-u)*(20):
> x75:=u*(3)+(1-u)*4: y75:=u*(-2)+(1-u)*0: z75:=u*20+(1-u)*(22):
> x76:=u*(4)+(1-u)*3: y76:=u*(0)+(1-u)*2: z76:=u*22+(1-u)*(20):
```

```
> d2:=plot3d([v*x74+(1-v)*x76,v*y74+(1-v)*y76, v*z74+(1-
  v)*z76],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> x77:=u*(3)+(1-u)*0: y77:=u*(2)+(1-u)*4: z77:=u*20+(1-u)*(20):
> x78:=u*(3)+(1-u)*2: y78:=u*(2)+(1-u)*4: z78:=u*20+(1-u)*(22):
> x79:=u*(2)+(1-u)*0: y79:=u*4+(1-u)*4: z79:=u*22+(1-u)*(20):
> d3:=plot3d([v*x77+(1-v)*x78,v*y77+(1-v)*y78, v*z77+(1-
  v)*z78],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> x80:=u*0+(1-u)*(-3): y80:=u*(4)+(1-u)*(2): z80:=u*20+(1-u)*(20):
> x81:=u*(0)+(1-u)*(-2): y81:=u*(4)+(1-u)*4: z81:=u*20+(1-u)*(22):
> x82:=u*(-2)+(1-u)*(-3): y82:=u*(4)+(1-u)*2: z82:=u*22+(1-
  u)*(20):
> d4:=plot3d([v*x80+(1-v)*x81,v*y40+(1-v)*y81, v*z80+(1-
  v)*z81],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> x83:=u*(-3)+(1-u)*(-3): y83:=u*(2)+(1-u)*(-2): z83:=u*20+(1-
  u)*(20):
> x84:=u*(-3)+(1-u)*(-4): y84:=u*(2)+(1-u)*0: z84:=u*20+(1-
  u)*(22):
> x85:=u*(-4)+(1-u)*(-3): y85:=u*(0)+(1-u)*(-2): z85:=u*22+(1-
  u)*(20):
> d5:=plot3d([v*x83+(1-v)*x84,v*y83+(1-v)*y84, v*z83+(1-
  v)*z84],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> x86:=u*(-3)+(1-u)*0: y86:=u*(-2)+(1-u)*(-4): z86:=u*20+(1-
  u)*(20):
> x87:=u*(-3)+(1-u)*(-2): y87:=u*(-2)+(1-u)*(-4): z87:=u*20+(1-
  u)*(22):
> x88:=u*(-2)+(1-u)*0: y88:=u*(-4)+(1-u)*(-4): z88:=u*22+(1-
  u)*(20):
> d6:=plot3d([v*x86+(1-v)*x87,v*y46+(1-v)*y87, v*z86+(1-
  v)*z87],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> k1:=plot3d([v*x52+(1-v)*x72,v*y52+(1-v)*y72, v*z52+(1-
  v)*z72],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> k2:=plot3d([v*x53+(1-v)*x73,v*y53+(1-v)*y73, v*z53+(1-
  v)*z73],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> k3:=plot3d([v*x55+(1-v)*x75,v*y55+(1-v)*y75, v*z55+(1-
  v)*z75],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> k4:=plot3d([v*x56+(1-v)*x76,v*y56+(1-v)*y76, v*z56+(1-
  v)*z76],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> k5:=plot3d([v*x58+(1-v)*x78,v*y58+(1-v)*y78, v*z58+(1-
  v)*z78],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> k6:=plot3d([v*x59+(1-v)*x79,v*y59+(1-v)*y39, v*z59+(1-
  v)*z79],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
```

```
> k7:=plot3d([v*x61+(1-v)*x81,v*y61+(1-v)*y81, v*z61+(1-
   v)*z81],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
   color="Pink", style=patchnogrid, lightmodel=light4):
> k8:=plot3d([v*x62+(1-v)*x82,v*y62+(1-v)*y82, v*z62+(1-
   v)*z82],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
   color="Pink", style=patchnogrid, lightmodel=light4):
> k9:=plot3d([v*x64+(1-v)*x84,v*y64+(1-v)*y84, v*z64+(1-
   v)*z84],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
   color="Pink", style=patchnogrid, lightmodel=light4):
> k10:=plot3d([v*x65+(1-v)*x85,v*y65+(1-v)*y85, v*z65+(1-
   v)*z85],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
   color="Pink", style=patchnogrid, lightmodel=light4):
> k11:=plot3d([v*x67+(1-v)*x87,v*y67+(1-v)*y87, v*z67+(1-
   v)*z87],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
   color="Pink", style=patchnogrid, lightmodel=light4):
> k12:=plot3d([v*x68+(1-v)*x88,v*y68+(1-v)*y88, v*z68+(1-
   v)*z88],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
   color="Pink", style=patchnogrid, lightmodel=light4):
>display(a1,a2,a3,a4,a5,a6,b1,b2,b3,b4,b5,b6,c1,c2,c3,c4,c5,c6,d1,
   d2,d3,d4,d5,d6,e4,e5,e6,e7,e8,e9,e10,e11,e12,k1,k2,k3,k4,k5,k6,
   k7,k8,k9,k10,k11,k12,s1,s2,s3,s4,s5,s6,s7,s8,s9,s10,s11,s12,axe
   s=normal,lightmodel=light3,scaling=constrained,labels=[x,y,z],
   color="Pink", style=patchnogrid, lightmodel=light4);
```

- **Model 3 (2 benda dasar bagian alas, 1 benda dasar bagian tengah, dan 2 benda dasar bagian atas)**

```
>a1:=plot3d([30*sin(v)*cos(u),30*sin(v)*sin(u),(30*cos(v))],u=30..
  2*Pi,v=Pi/3..3*Pi/4,scaling=constrained, labels=[x,y,z],
  color="Violet", style=patchnogrid, lightmodel=light4):
> tcemb1:=0: tcemb2:=10:tcemb3:=30: #ketinggian titik kontrol#
kcb:=2: #faktor pengali dilatasi segi-6#
rcemb1:=25: rcemb2:=35: #titik kontrol pd sb x&y#

for k from 0 to 5 do
ccb[2*k+1]:="Violet": ccb[2*k+2]:="Violet":
c1[k+1]:=plot3d([(1-v)*((1-u)^2*rcemb1+2*(1-
  u)*u*rcemb2+u^2*rcemb1)*cos(Pi/3*k)+v*((1-u)^2*rcemb1+2*(1-
  u)*u*rcemb2+u^2*rcemb1)*cos(Pi/3*(k+1)),(1-v)*((1-
  u)^2*rcemb1+2*(1-u)*u*rcemb2+u^2*rcemb1)*sin(Pi/3*k)+v*((1-
  u)^2*rcemb1+2*(1-u)*u*rcemb2+u^2*rcemb1)*sin(Pi/3*(k+1)),(1-
  u)^2*tcemb1+2*(1-u)*u*tcemb2+u^2*tcemb3+(-
  47)],u=0..1,v=0..1,color=ccb[k+1]):
end do:
a2:=display({c1[1],c1[2],c1[3],c1[4],c1[5],c1[6]},scaling=constrai
  ned, labels=[x,y,z], color="Violet", style=patchnogrid,
  lightmodel=light4):
>a3:=plot3d([(26*(1-t)^2+15*2*(1-t)*t+t^2*15)*cos(v),(26*(1-
  t)^2+15*2*(1-t)*t+t^2*15)*sin(v),(14.5*(1-t)^2+14.5*2*(1-
  t)*t+t^2*14.5)],t=0..1,v=0..2*Pi,scaling=constrained,
  labels=[x,y,z], color="Violet", style=patchnogrid,
  lightmodel=light4):
>a4:=plot3d([(18*(1-t)^2+15*2*(1-t)*t+t^2*18)*cos(v),(18*(1-
  t)^2+15*2*(1-t)*t+t^2*18)*sin(v),(12.5*(1-t)^2+18*2*(1-
```

```
  t)*t+t^2*20)],t=0..1,v=0..2*Pi,scaling=constrained,
  labels=[x,y,z], color="Violet", style=patchnogrid,
  lightmodel=light4):
>a5:=plot3d([(18*(1-t)^2+20*2*(1-t)*t+t^2*18)*cos(v),(18*(1-
  t)^2+20*2*(1-t)*t+t^2*18)*sin(v),(20*(1-t)^2+25*2*(1-
  t)*t+t^2*30)],t=0..1,v=0..2*Pi,scaling=constrained,
  labels=[x,y,z], color="Violet", style=patchnogrid,
  lightmodel=light4):
>a6:=plot3d([(18*(1-t)^2+20*2*(1-t)*t+t^2*18)*cos(v),(18*(1-
  t)^2+20*2*(1-t)*t+t^2*18)*sin(v),(30*(1-t)^2+35*2*(1-
  t)*t+t^2*40)],t=0..1,v=0..2*Pi,scaling=constrained,
  labels=[x,y,z], color="Violet", style=patchnogrid,
  lightmodel=light4):
>a7:=plot3d([(18*(1-t)^2+15*2*(1-t)*t+t^2*10)*cos(v),(18*(1-
  t)^2+15*2*(1-t)*t+t^2*10)*sin(v),(40*(1-t)^2+40*2*(1-
  t)*t+t^2*40)],t=0..1,v=0..2*Pi,scaling=constrained,
  labels=[x,y,z], color="Violet", style=patchnogrid,
  lightmodel=light4):
>a8:=plot3d([18*cos(u),18*sin(u),v+40],u=0..Pi/4,v=0..200,scaling=
  constrained, labels=[x,y,z], color="Pink", style=patchnogrid,
  lightmodel=light4):
>a9:=rotate(a8,0,0,Pi/2):a10:=rotate(a8,0,0,Pi/2):a11:=rotate(a8,0
  ,0,Pi):
>a12:=plot3d([15*cos(u),15*sin(u),v+40],u=0..Pi/4,v=0..200,scaling
  =constrained, labels=[x,y,z], color="Pink", style=patchnogrid,
  lightmodel=light4):
>a13:=rotate(a12,0,0,Pi/2):a14:=rotate(a12,0,0,-
  Pi/2):a15:=rotate(a12,0,0,Pi):
>a16:=plot3d([15*cos(u),15*sin(u),v+40],u=0..2*Pi,v=0..200,scaling
  =constrained, labels=[x,y,z], color="Pink", style=patchnogrid,
  lightmodel=light4):
> x9:=u*(-18) +(1-u)*(-15):       y9:=u*0 +(1-u)*0: z9:=u*40 +(1-
  u)*(40):
> x10:=u*(-18) +(1-u)*(-15): y10:=u*0 +(1-u)*0: z10:=u*240 +(1-
  u)*(240):
>a17:=plot3d([v*x9+(1-v)*x10,v*y9+(1-v)*y10,v*z9+(1-
  v)*z10],u=0..1,v=0..1,scaling=constrained,labels=[x,y,z],
  color="Pink", style=patchnogrid, lightmodel=light4):
> a18:=rotate(a17,0,0,Pi/2):
> a19:=rotate(a17,0,0,Pi/4):
> a20:=rotate(a17,0,0,2*Pi/2.66):
> a21:=rotate(a17,0,0,Pi):
> a22:=rotate(a17,0,0,5*Pi/4):
> a23:=rotate(a17,0,0,3*Pi/2):
> a24:=rotate(a17,0,0,7*Pi/4):
>a25:=plot3d([(18*(1-t)^2+15*2*(1-t)*t+t^2*10)*cos(v),(18*(1-
  t)^2+15*2*(1-t)*t+t^2*10)*sin(v),(240*(1-t)^2+240*2*(1-
  t)*t+t^2*240)],t=0..1,v=0..2*Pi,scaling=constrained,
  labels=[x,y,z], color="Violet", style=patchnogrid,
  lightmodel=light4):
>a26:=plot3d([(18*(1-t)^2+15*2*(1-t)*t+t^2*18)*cos(v),(18*(1-
  t)^2+15*2*(1-t)*t+t^2*18)*sin(v),(240*(1-t)^2+250*2*(1-
  t)*t+t^2*250)],t=0..1,v=0..2*Pi,scaling=constrained,
  labels=[x,y,z], color="Violet", style=patchnogrid,
  lightmodel=light4):
```

```
>a27:=plot3d([(18*(1-t)^2+20*2*(1-t)*t+t^2*18)*cos(v),(18*(1-
  t)^2+20*2*(1-t)*t+t^2*18)*sin(v),(250*(1-t)^2+260*2*(1-
  t)*t+t^2*270)],t=0..1,v=0..2*Pi,scaling=constrained,
  labels=[x,y,z], color="Violet", style=patchnogrid,
  lightmodel=light4):
>a28:=plot3d([(18*(1-t)^2+20*2*(1-t)*t+t^2*18)*cos(v),(18*(1-
  t)^2+20*2*(1-t)*t+t^2*18)*sin(v),(270*(1-t)^2+280*2*(1-
  t)*t+t^2*290)],t=0..1,v=0..2*Pi,scaling=constrained,
  labels=[x,y,z], color="Violet", style=patchnogrid,
  lightmodel=light4):
>a29:=plot3d([30*sin(v)*cos(u),30*sin(v)*sin(u),(30*cos(v)+306)],u
  =40..2*Pi,v=Pi/3..3*Pi/4,scaling=constrained, labels=[x,y,z],
  color="Violet", style=patchnogrid, lightmodel=light4):
>a30:=plot3d([30*sin(v)*cos(u),30*sin(v)*sin(u),(30*cos(v)+337)],u
  =40..2*Pi,v=Pi/3..3*Pi/4,scaling=constrained, labels=[x,y,z],
  color="Violet", style=patchnogrid, lightmodel=light4):
>a31:=plot3d([(21*(1-t)^2+15*2*(1-t)*t+t^2*10)*cos(v),(21*(1-
  t)^2+15*2*(1-t)*t+t^2*10)*sin(v),(285*(1-t)^2+285*2*(1-
  t)*t+t^2*285)],t=0..1,v=0..2*Pi,scaling=constrained,
  labels=[x,y,z], color="Violet", style=patchnogrid,
  lightmodel=light4):
>display(a1,a2,a3,a4,a5,a6,a7,a8,a9,a10,a11,a12,a13,a14,a15,a16,a1
  7,a18,a19,a20,a21,a22,a23,a24,a25,a26,a27,a28,a29,a30,a31,axes=
  normal);
```