> alias(beta_plus = sqrt(2 * sqrt(d^2 - 10 * d + 9) + 2 * d - 6), beta_minus = sqrt(- 2 * sqrt(d^2 - 10 * d + 9) + 2 * d - 6))

$$beta\_plus,\ beta\_minus \quad (1)$$

> $alias\left(eta\_plus = d + 3 + \mathrm{sqrt}\left(d^2 - 10\cdot d + 9\right), eta\_minus = d + 3 - \mathrm{sqrt}\left(d^2 - 10\cdot d + 9\right)\right)$

$$beta\_plus,\ beta\_minus,\ eta\_plus,\ eta\_minus \quad (2)$$

> $J\_appx := (v, x, n) \rightarrow sum\left(\frac{\left(\frac{x}{2}\right)^v \cdot (-1)^k}{k!\cdot \mathrm{GAMMA}(k + v + 1)} \cdot \left(\frac{x^2}{4}\right)^k, k = 0..n\right)$

$$J\_appx := (v, x, n) \mapsto \sum_{k=0}^{n} \frac{\left(\frac{x}{2}\right)^v \cdot (-1)^k \cdot \left(\frac{x^2}{4}\right)^k}{k!\cdot \Gamma(k + v + 1)} \quad (3)$$

> $error\_bar := (x, n, v) \rightarrow \frac{\left|\frac{x}{2}\right|^{2n + v}}{\mathrm{factorial}(n)\cdot \mathrm{GAMMA}(n + v + 1)\cdot\left(1 - \frac{|x|^2}{(2n + 2)^2}\right)}$

$$error\_bar := (x, n, v) \mapsto \frac{\left|\frac{x}{2}\right|^{2\cdot n + v}}{n!\cdot \Gamma(n + v + 1)\cdot\left(1 - \frac{|x|^2}{(2\cdot n + 2)^2}\right)} \quad (4)$$

> $d := 8$

$$d := 8 \quad (5)$$

> $n := 5$

$$n := 5 \quad (6)$$

> $numerator\_appx := \mathrm{Im}\left(\frac{eta\_minus\cdot beta\_minus}{\mathrm{GAMMA}\left(\frac{d}{2}\right)} \cdot \left(\frac{\mathrm{lambda}\cdot beta\_minus}{4}\right)^{\frac{d}{2} - 1} \cdot J\_appx\left(\frac{d}{2}, \frac{\mathrm{lambda}\cdot beta\_plus}{2}, n\right)\right)$

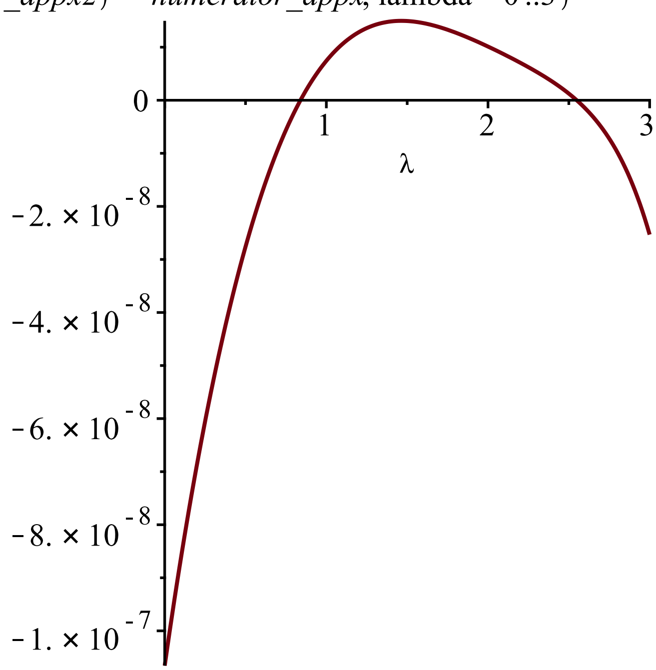
$$numerator\_appx := \frac{1}{384}\left(\Im\left(\left(11 - \sqrt{-7}\right)\left(-2\sqrt{-7} + 10\right)^2 \lambda^3 \left(\frac{\lambda^4 \left(2\sqrt{-7} + 10\right)^2}{6144} - \frac{\lambda^6 \left(2\sqrt{-7} + 10\right)^3}{491520} + \frac{\lambda^8 \left(2\sqrt{-7} + 10\right)^4}{94371840} - \frac{\lambda^{10} \left(2\sqrt{-7} + 10\right)^5}{31708938240} + \frac{\lambda^{12} \left(2\sqrt{-7} + 10\right)^6}{16234976378880} - \frac{\lambda^{14} \left(2\sqrt{-7} + 10\right)^7}{11689182992793600}\right)\right)\right) \quad (7)$$

>

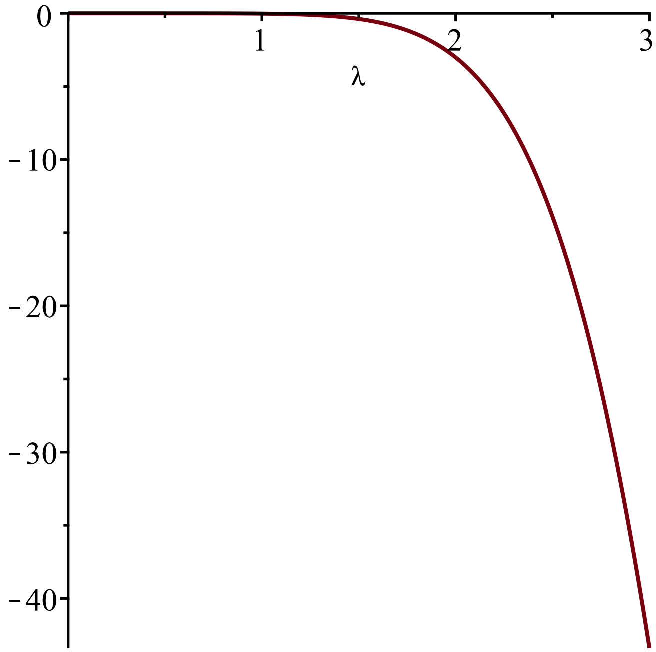
> $numerator\_appx2 := mtaylor\left(\frac{eta\_minus \cdot beta\_minus}{\mathrm{GAMMA}\left(\frac{d}{2}\right)} \cdot \left(\frac{\mathrm{lambda} \cdot beta\_minus}{4}\right)^{\frac{d}{2}-1} \cdot J\_appx\left(\frac{d}{2}, \frac{\mathrm{lambda} \cdot beta\_plus}{2}, n\right), \mathrm{lambda} = 2.5, 2\,n + d\right)$

$$numerator\_appx2 := 15.96482665 - 13.90880791\,\mathrm{I} + (1.907815845\;10^{-7} - 2.635178297\;10^{-7}\,\mathrm{I})\,(\lambda - 2.5)^{16} + (3.747680295\;10^{-6} - 4.787969889\;10^{-6}\,\mathrm{I})\,(\lambda - 2.5)^{15} + (4.488978462\;10^{-9} - 6.200419522\;10^{-9}\,\mathrm{I})\,(\lambda - 2.5)^{17} + (27.56381633 - 36.68494818\,\mathrm{I})\,(\lambda - 2.5) + (0.03006943800 - 0.07108546985\,\mathrm{I})\,(\lambda - 2.5)^{9} + (7.404808001 - 37.67544206\,\mathrm{I})\,(\lambda - 2.5)^{2} - (15.42252310 + 15.85138231\,\mathrm{I})\,(\lambda - 2.5)^{3} + (0.01320489827 - 0.01268269248\,\mathrm{I})\,(\lambda - 2.5)^{10} + (0.9286973663 - 0.06383211824\,\mathrm{I})\,(\lambda - 2.5)^{7} + (-14.10625936 + 1.650902430\,\mathrm{I})\,(\lambda - 2.5)^{4} + (0.0003652494654 - 0.0002806127052\,\mathrm{I})\,(\lambda - 2.5)^{13} + (-3.289228941 + 4.485589943\,\mathrm{I})\,(\lambda - 2.5)^{5} + (1.256077579 + 1.586212924\,\mathrm{I})\,(\lambda - 2.5)^{6} + (0.2150979045 - 0.2133964407\,\mathrm{I})\,(\lambda - 2.5)^{8} + (0.006772431621 - 0.002680960328\,\mathrm{I})\,(\lambda - 2.5)^{11} + (0.00004514721873 - 0.00004778995600\,\mathrm{I})\,(\lambda - 2.5)^{14} + (0.001986088938 - 0.0009912723233\,\mathrm{I})\,(\lambda - 2.5)^{12} \quad \textbf{(8)}$$

> $plot(\mathrm{Im}(numerator\_appx2) - numerator\_appx, \mathrm{lambda} = 0..3)$



> $plot(numerator\_appx, \text{lambda} = 0..3)$

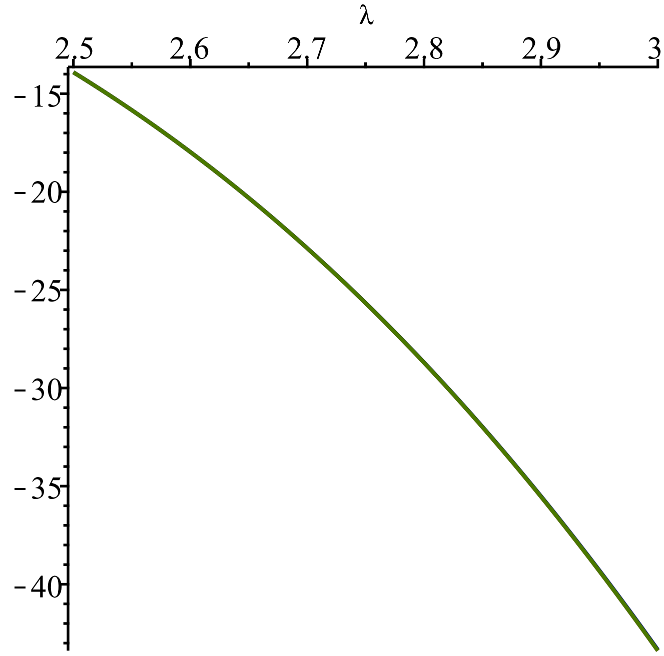


>

> $numerator := \mathrm{Im}\left(\frac{eta\_minus \cdot beta\_minus}{\mathrm{GAMMA}\left(\frac{d}{2}\right)} \cdot \left(\frac{\text{lambda} \cdot beta\_minus}{4} \right)^{\frac{d}{2}-1} \cdot \mathrm{BesselJ}\left(\frac{d}{2}, \frac{\text{lambda} \cdot beta\_plus}{2} \right) \right)$

$$numerator := \frac{\Im\left(\left(11 - \sqrt{-7}\right) \left(-2\sqrt{-7} + 10\right)^2 \lambda^3 \,\mathrm{BesselJ}\left(4, \frac{\lambda \sqrt{2\sqrt{-7} + 10}}{2}\right) \right)}{384} \tag{9}$$

> $plot(\{numerator, numerator\_appx, \mathrm{Im}(numerator\_appx2)\}, \text{lambda} = 2.5..3)$




```
> 
```

```
> poly_val := evalf(subs(lambda = 2.5, numerator_appx))
```

$$poly\_val := -13.90880791 \tag{10}$$

```
> error_appx := evalf(|eta_minus·beta_minus/GAMMA(d/2) · (3·beta_minus/4)^(d/2 - 1)| ·error_bar(3·beta_plus/2, n, d/2))
```

$$error\_appx := \text{evalf}\left(\left|\frac{eta\_minus \cdot beta\_minus}{\text{GAMMA}\left(\frac{d}{2}\right)} \cdot \left(\frac{3 \cdot beta\_minus}{4}\right)^{\frac{d}{2}-1}\right| \cdot error\_bar\left(\frac{3 \cdot beta\_plus}{2}, n, \frac{d}{2}\right)\right)$$

$$error\_appx := 1.200820628 \tag{11}$$

```
> with(plots)
```

[*animate, animate3d, animatecurve, arrow, changecoords, complexplot, complexplot3d, conformal, conformal3d, contourplot, contourplot3d, coordplot, coordplot3d, densityplot, display, dualaxisplot, fieldplot, fieldplot3d, gradplot, gradplot3d, implicitplot, implicitplot3d, inequal, interactive, interactiveparams, intersectplot, listcontplot, listcontplot3d, listdensityplot, listplot, listplot3d, loglogplot, logplot, matrixplot, multiple, odeplot, pareto, plotcompare, pointplot, pointplot3d, polarplot, polygonplot, polygonplot3d, polyhedra_supported, polyhedraplot, rootlocus, semilogplot, setcolors, setoptions, setoptions3d, shadebetween, spacecurve, sparsematrixplot, surfdata, textplot, textplot3d, tubeplot*] **(12)**

```
> numerator_appx_func :=proc(d, n)
  local numerator, numerator_appx, eta_minus, beta_minus, beta_plus, eta_plus, J_appx,
      error_bar, error_appx, poly_upperbound;
  beta_plus := sqrt(2·sqrt(d) + d − 3) + I·sqrt(2·sqrt(d) − d + 3);
  beta_minus := sqrt(2·sqrt(d) + d − 3) - I·sqrt(2·sqrt(d) − d + 3);
  eta_plus := d + 3 + sqrt(d^2 − 10·d + 9);
```

$$eta\_minus := d + 3 - \mathrm{sqrt}\left(d^2 - 10 \cdot d + 9\right);$$

$$J\_appx := (v, x, n) \rightarrow sum\left(\frac{\left(\frac{x}{2}\right)^{v} \cdot (-1)^{k}}{k! \cdot \mathrm{GAMMA}(k + v + 1)} \cdot \left(\frac{x^2}{4}\right)^{k}, k = 0..n\right);$$

$$error\_bar := (x, n, v) \rightarrow \frac{\left|\frac{x}{2}\right|^{2n+v}}{\mathrm{factorial}(n) \cdot \mathrm{GAMMA}(n + v + 1) \cdot \left(1 - \frac{|x|^2}{(2n+2)^2}\right)};$$

$$numerator := \mathrm{Im}\left(\frac{eta\_minus \cdot beta\_minus}{\mathrm{GAMMA}\left(\frac{d}{2}\right)} \cdot \left(\frac{\mathrm{lambda} \cdot beta\_minus}{4}\right)^{\frac{d}{2} - 1} \cdot \mathrm{BesselJ}\left(\frac{d}{2}, \frac{\mathrm{lambda} \cdot beta\_plus}{2}\right)\right);$$

$$numerator\_appx := \mathrm{Im}\left(\frac{eta\_minus \cdot beta\_minus}{\mathrm{GAMMA}\left(\frac{d}{2}\right)} \cdot \left(\frac{\mathrm{lambda} \cdot beta\_minus}{4}\right)^{\frac{d}{2} - 1} \cdot J\_appx\left(\frac{d}{2}, \frac{\mathrm{lambda} \cdot beta\_plus}{2}, n\right)\right);$$

$$error\_appx := evalf\left(\left|\frac{eta\_minus \cdot beta\_minus}{\mathrm{GAMMA}\left(\frac{d}{2}\right)} \cdot \left(\frac{3 \cdot beta\_minus}{4}\right)^{\frac{d}{2} - 1}\right| \cdot error\_bar\left(\frac{3 \cdot beta\_plus}{2}, n, \frac{d}{2}\right)\right);$$

poly_upperbound := *evalf*(*subs*(lambda = 2.5, *numerator_appx*));
printf("d=%g \n", *d*);
printf("the error bounds of polynomail approximation= %f\n", *error_appx*);
printf("the truncated polynomail of the numerator at lambda 2.5= %f\n", *poly_upperbound*);
printf("the numerator at lambda 2.5= %f \n", *evalf*(*subs*(lambda = 2.5, *numerator*)));
plot({*numerator*, *numerator_appx*}, lambda = 2.5 ..3)

end proc

numerator_appx_func := **proc**(*d*, *n*) **(13)**

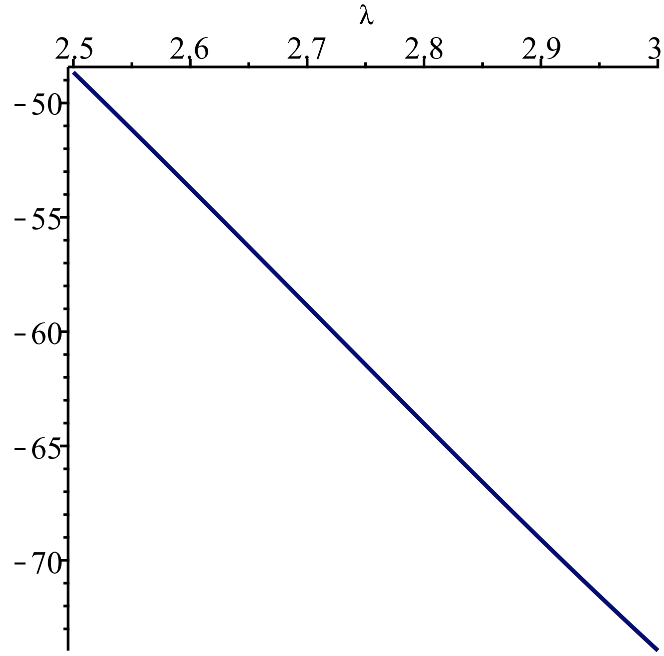
```
local numerator, numerator_appx, eta_minus, beta_minus, beta_plus, eta_plus, J_appx,
error_bar, error_appx, poly_upperbound;
beta_plus := sqrt(2 * sqrt(d) + d - 3) + I* sqrt(2 * sqrt(d) - d + 3);
beta_minus := sqrt(2 * sqrt(d) + d - 3) - I* sqrt(2 * sqrt(d) - d + 3);
eta_plus := d + 3 + sqrt(d^2 - 10 * d + 9);
eta_minus := d + 3 - sqrt(d^2 - 10 * d + 9);
J_appx := (v, x, n) → sum((1/2 * x)^v * (-1)^k * (1/4 * x^2)^k/(factorial(k)
 * GAMMA(k + v + 1)), k = 0 ..n);
error_bar := (x, n, v) → abs(1/2 * x)^(2 * n + v)/(factorial(n) * GAMMA(n + v + 1)
 * (1 - abs(x)^2/(2 * n + 2)^2));
numerator := Im(beta_minus * eta_minus * (1/4 * lambda * beta_minus)^(1/2 * d - 1)
 * BesselJ(1/2 * d, 1/2 * lambda * beta_plus)/GAMMA(1/2 * d));
numerator_appx := Im(beta_minus * eta_minus * (1/4 * lambda * beta_minus)^(1/2
 * d - 1) * J_appx(1/2 * d, 1/2 * lambda * beta_plus, n)/GAMMA(1/2 * d));
error_appx := evalf(abs(beta_minus * eta_minus * (3/4 * beta_minus)^(1/2 * d - 1)
 /GAMMA(1/2 * d)) * error_bar(3/2 * beta_plus, n, 1/2 * d));
poly_upperbound := evalf(subs(lambda = 2.5, numerator_appx));
printf("d=%g \n", d);
printf("the error bounds of polynomail approximation= %f\n", error_appx);
printf("the truncated polynomail of the numerator  at lambda 2.5= %f\n",
poly_upperbound);
printf("the numerator  at lambda 2.5= %f \n", evalf(subs(lambda = 2.5, numerator)));
plot({numerator, numerator_appx}, lambda = 2.5 ..3)
end proc
```

```
>
```

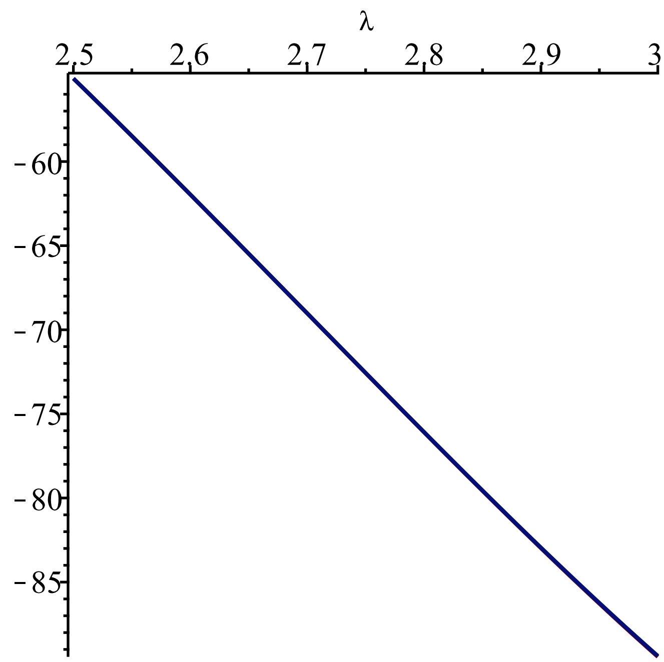
```
> n := 5;
  for d from 3 by 1 to 8 do
  numerator_appx_func(d, n)
   end do;
```

$$n := 5$$

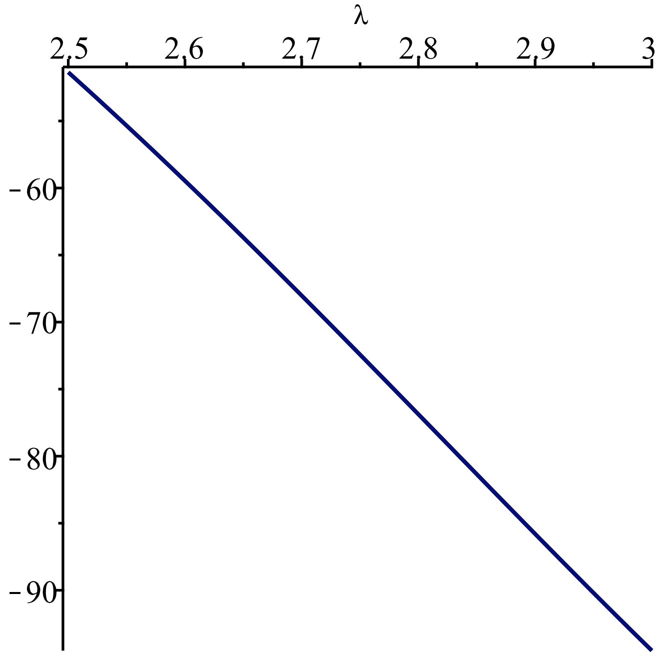
```
d=3
the error bounds of polynomail approximation= 0.359989
the truncated polynomail of the numerator  at lambda 2.5=
-48.656672
the numerator  at lambda 2.5= -48.655506
```



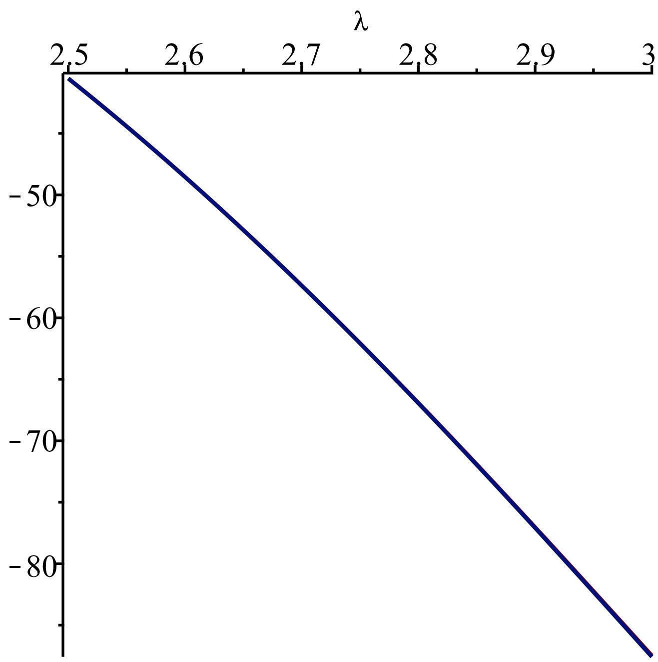
d=4
the error bounds of polynomail approximation= 0.753176
the truncated polynomail of the numerator at lambda 2.5=
-55.063129
the numerator at lambda 2.5= -55.059652



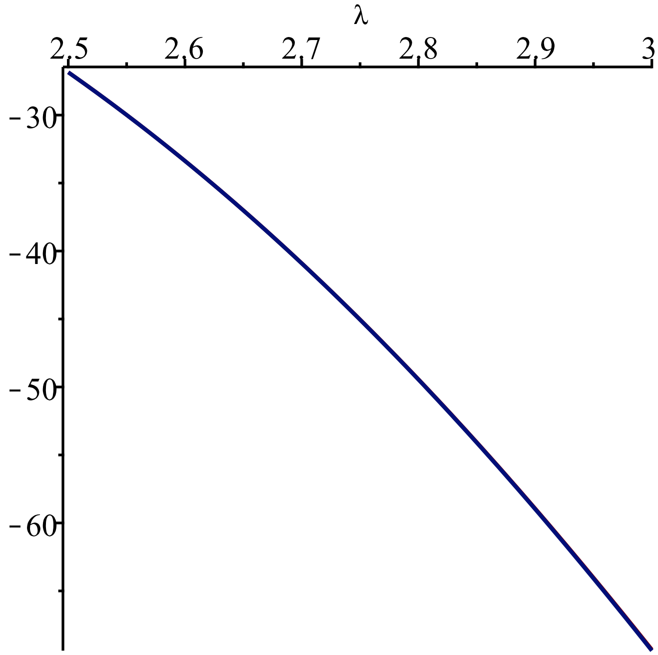
d=5
the error bounds of polynomail approximation= 1.133329
the truncated polynomail of the numerator at lambda 2.5=
-51.368007
the numerator at lambda 2.5= -51.368767



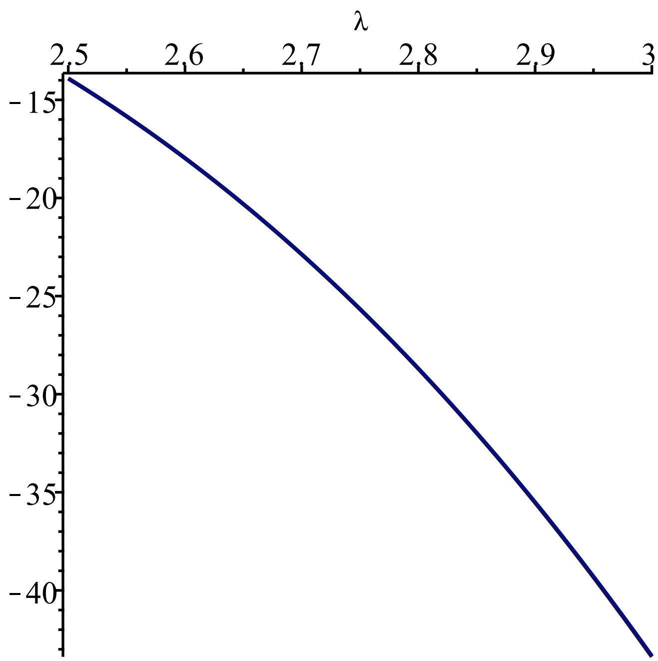
d=6
the error bounds of polynomail approximation= 1.356976
the truncated polynomail of the numerator at lambda 2.5=
-40.528560
the numerator at lambda 2.5= -40.533430



d=7
the error bounds of polynomail approximation= 1.367401
the truncated polynomail of the numerator at lambda 2.5=
-26.851665
the numerator at lambda 2.5= -26.854657



d=8
the error bounds of polynomail approximation= 1.200821
the truncated polynomail of the numerator at lambda 2.5=
-13.908808
the numerator at lambda 2.5= -13.907357



>