

### 3. 積分変換による数式処理

#### 3.1 ラプラス変換による微分方程式の解

```
> restart;  
with(inttrans):
```

微分方程式の定義

```
> eqa:=2*diff(y(t),t)+6*y(t)+4*int(y(x),x=0..t)=20*exp(-3*t);
```

ラプラス変換

y(t) をラプラス変換したときの表示を指定しておきます。

```
> addtable(laplace,y(t),Y(s),t,s);  
eqa2:=laplace(eqa,t,s);
```

ラプラス変換した式を Y(s) について解きます。

```
> eqa3:=solve({subs(y(0)=0,eqa2)},{Y(s)});
```

逆ラプラス変換

Y(s) を逆ラプラス変換したときの表示を指定します。

```
> addtable(invlaplace,Y(s),y(t),s,t);  
eqa4:=invlaplace(eqa3,s,t);
```

解のプロット

得られた解を関数に書き直します。

```
> y:=unapply(subs(eqa4,y(t)),t);  
plot(y(t),t=0..10);
```

#### 3.2 逆ラプラス変換によるシステムの過渡応答

伝達関数 G で表されるシステムの過渡応答 (時間応答) を求めます。

システムの伝達関数

```
> restart;  
G:=(1+2*s)/(1+3*s+2.5*s^2+1.2*s^3+0.3*s^4);
```

ステップ応答

```
> stepG:=G*(1/s);
```

逆ラプラス変換を行い、時間応答を求めます。

```
> with(inttrans):  
> stepT:=invlaplace(stepG,s,t);  
> stepT:=simplify(stepT);
```

プロット

```
> plot(stepT,t=0..20,title="ステップ応答");
```

インパルス応答

```
> impulseG:=G*1;  
> impulseT:=invlaplace(impulseG,s,t);  
> impulseT:=simplify(impulseT);
```

プロット

```
> plot(impulseT,t=0..20,title="インパルス応答");
```