```
syms S kp ki kd % S is the laplace transform s
```

```
G_sym = 15000/(S^4+50*S^3+875*S^2+6250*S+15000); % plant
C_sym = kp + ki/S + kd*S; % PID controller transfer fun
```

```
CL_sym = simplify( ...
    C_sym*G_sym/(1+C_sym*G_sym) ...
    )
```

CL_sym =

$$\frac{15000\,\mathrm{kd}\,S^2 + 15000\,\mathrm{kp}\,S + 15000\,\mathrm{ki}}{15000\,S + 15000\,\mathrm{ki} + 15000\,S\,\mathrm{kp} + 15000\,S^2\,\mathrm{kd} + 6250\,S^2 + 875\,S^3 + 50\,S^4 + S^5}$$

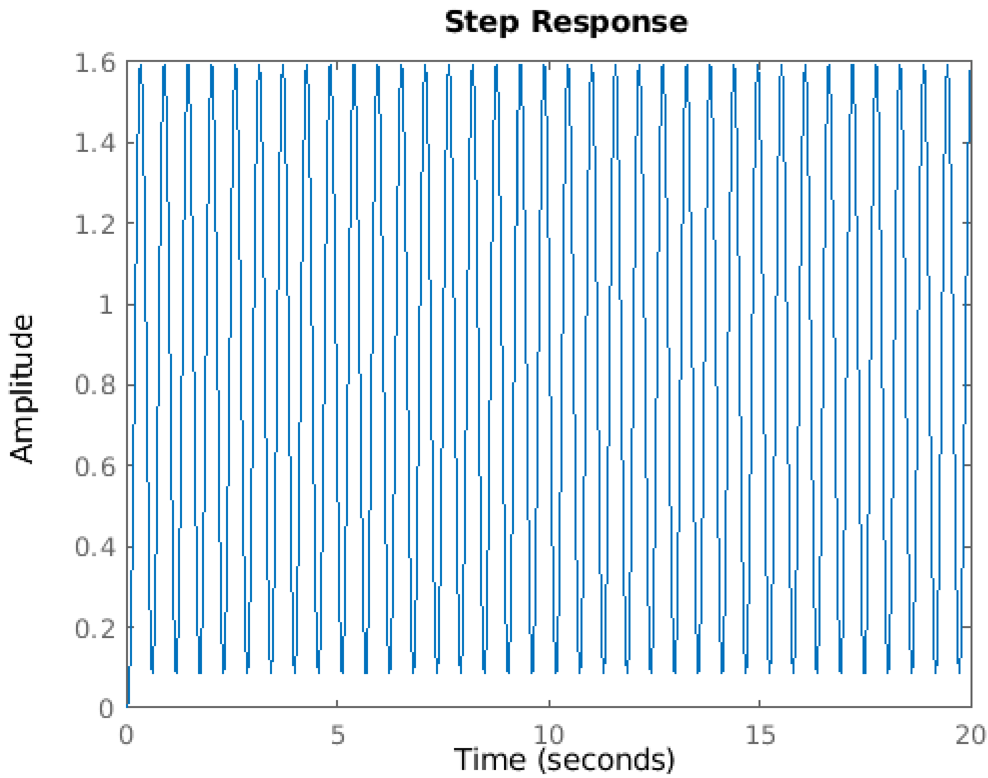
```
K_sub = @(Kp,Ki,Kd) sym_to_tf( ...
    subs( ...
        CL_sym, ...
        {kp,ki,kd}, ...
        {Kp,Ki,Kd} ...
        ), ...
        S ...
    );
K_sub(1,0,0) % e.g.
```

```
ans =

                   15000 s
  ------------------------------------------
  s^5 + 50 s^4 + 875 s^3 + 6250 s^2 + 30000 s

Continuous-time transfer function.
```

```
step(K_sub(5.25, 0, 0))
```




```
Ku = 5.25
```

```
Ku = 5.2500
```

```
Tu = 5 / 9
```

```
Tu = 0.5556
```

```
Kp = 0.6 * Ku
```

```
Kp = 3.1500
```

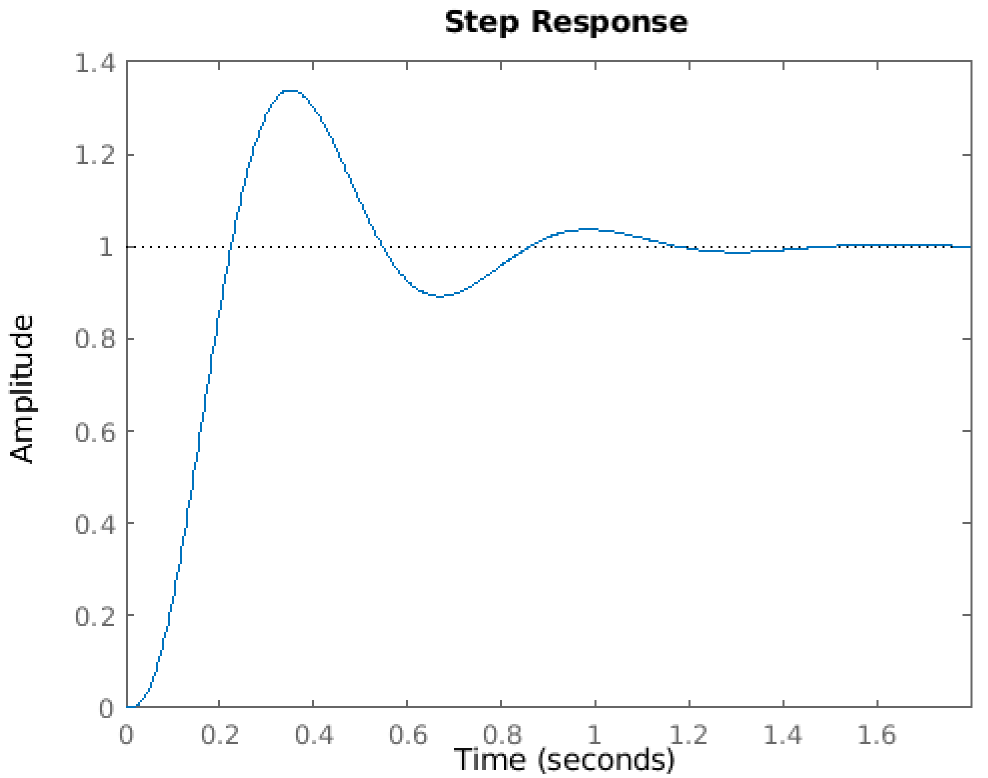
```
Ki = 1.2 * Ku / Tu
```

```
Ki = 11.3400
```

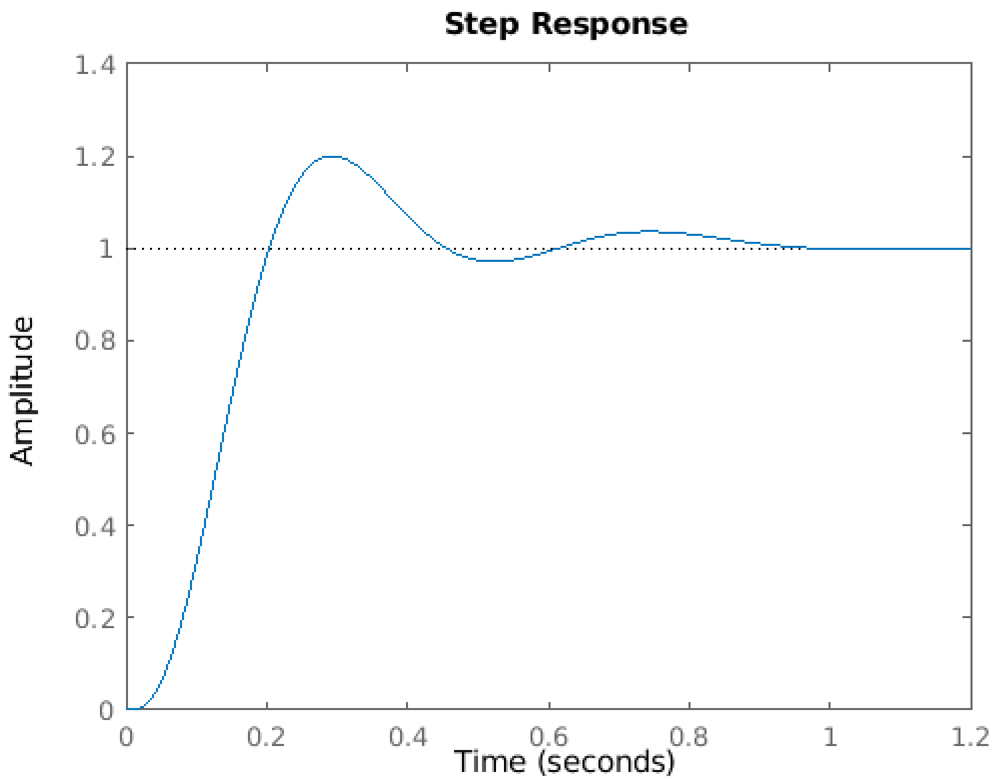
```
Kd = 3 * Ku * Tu / 40
```

```
Kd = 0.2188
```

```
step(K_sub(Kp, Ki, Kd))
```




```
step(K_sub(Kp, Ki, 0.35))
```



```
function tf_obj = sym_to_tf(sym_tf,s_var)
    % TODO test to make sure s_var is in,→ symvar(sym_tf) ...
    syms(symvar(sym_tf))
    syms s
    sym_tf = subs(sym_tf,s_var,s);
    tf_str = char(sym_tf);
    s = tf([1,0],[1]);
    eval(['tf_obj = ',tf_str,';']);
end
```