UBC ELECTRICAL ENGINEERING DEPARTMENT

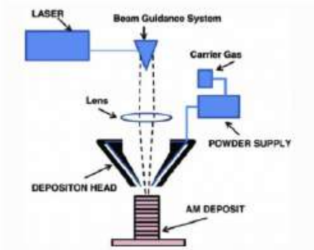
Temperature control system of Direct Laser Deposition for additive manufacturing

Background

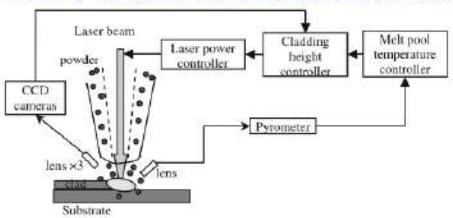
Direct Laser Deposition (DLD) has become a widely used Laser-based additive manufacturing method in the area of 3D printing in the last few years. A big challenge is to build a temperature control system to ensure quality and reliability of the components.

Introduction

Additive Manufacturing (AM) fabricates parts layer by layer from 3D model. DLD is a specific technique used in AM that use laser beam to melt the powder of the selected materials to produce the parts. The working principle is that the laser beam is used as the heat source and metallic powder is fed into the molten pool. With the movement of the deposition head, the solidified molten pool will create a new layer onto the AM Deposit.



The figure below shows a simple control setup of the AM system. In this research, we only focus on the temperature control.



Melt Pool Temperature Control

Input Signal

Temperature is the input signal of the system. The temperature adjusted by the controller varies with the voltage. To get the real-time disturbance, we need to subtract the desired voltage and the current temperature of the melt pool. There are two types of pyrometers to measure the current temperature:

- Contact:

It measures the thermo-electric electromotive force generated by the temperature gradient at different ends to transfer temperature signals into electrical signals.

- Non-Contact:

It uses infrared to detect the temperature. By Planck's law, we know that the energy emission can be expressed as below. Knowing the emissivity Lr, we can calculate thermal signal Tb

$$L_r(T_B, \lambda) = \epsilon L(T_B, \lambda) = \epsilon \frac{C_1}{\lambda^5 [e^{(C_2/\lambda T_B)} - 1]}$$

Output Signal

The output signal is voltage. After setting a desired temperature, starting from 0, the input voltage into this control system will start to modify itself generation by generation. Note that the voltage is the attribute that affects temperature of the melt pool.

Control Parameters

The dynamic model is identified using second order state-space model with the form

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k)$$

$$\mathbf{y}(k) = \mathbf{C}x(k) + \mathbf{D}\mathbf{u}(k)$$

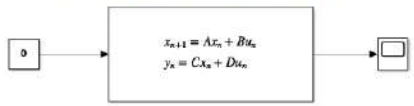
Control Parameters Continued

- x: state vector
- y: output, temperature
- u: input, voltage
- A, B, C, and D are the matrices defining the state space model, defined as

```
A = [0.98507 -0.01655 0.016227 0.00011625;
     0.019185 0.96107 0.21983 0.036113;
     -0.021373 0.012355 0.33864 -0.72304;
     0.0075648 0.028016 -0.42702 -0.65477 ];
B = [0.0005367;
     -0.015565;
     0.027322;
     -0.012322];
C = 1.8*[7648.1 26.713 -330.72 -17.855];
D = 0;
```

Control Process

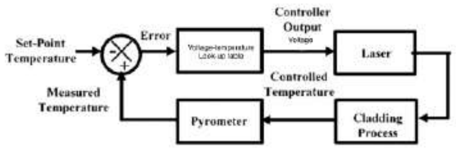
The control system design is based on a feedforward control loop. The first step is to generate a voltage-temperature lookup table using the following design diagram.



Since every voltage has its unique temperature output, when we get a temperature output, we can inversely find the voltage input.

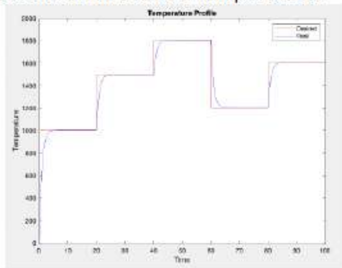
For example, when we have 1.42V, the temperature output is 2191.2℃, vice versa.

To start with temperature controller, we first need to set the temperature we desire into the control model. The initial voltage state is set as 0. The figure below is the control diagram we build.



Control Process Continued

By getting the difference between desired temperature and measured temperature, we iteratively run the control model and settle down the device at its desired temperature.



The above figure shows how temperature desired to change and in real. We can see there will always be delay to change from one temperature to another. Code added below as reference.

```
sys = ss(A,B,C,D,dt);
l=1;
for vol = 0:0.01:2
    v = [vol*ones(1,N)]; % Voltage value from 0.8 - 1.6v
    yt = lsim(sys,v,0:dt:(N-1)*dt);
    table(l,:) = [vol,yt(end)] ;
    l=l+1;
end
tem_table = table(:,2);
vol_table = ref;
for i = 1:size(ref,1)
    tem = ref(i,2);
    [~,ia,~]=unique(round(abs(tem_table-tem)),'sorted');
    vol_table(i,2)=table(ia(1),1);
end
u = vol_table(:,2);
x = [0;0;0;0]; % initial conditions
for i = 1:size(ref,1)
        y(:,i) = C*x+ D*u(i);
        x = A*x + B*u(i);
end
figure
plot(ref(:,1),ref(:,2),color="red")
hold on
plot(ref(:,1),y,Color="blue")
xlabel('Time')
ylabel('Temperature')
title('Temperature Profile')
legend('Desired', 'Real');
```

Conclusion

DLD has a bright and wide application scenarios in the future. With feedforward control system, the DLD will self-correct and have less flaws and defects on the produced parts. This poster shows a possible method of achieving it.

THE UNIVERSITY OF BRITISH COLUMBIA

Mentor Kezi Li