1. BACKGROUND

One difficulty with metal additive manufacturing is the requirement to balance the laser power and travel speed to avoid defects. Too much heat (high power, low speed) and the material reaches boiling point leading to keyhole melting and trapped gas pores; too little heat (low power, high speed) and the material does not join together leading to ‘lack-of-fusion’ pores. Porosity is undesirable because it will cause the material to have reduced mechanical properties, which may result in the component failing prematurely in operation.

While it is relatively straightforward to optimise the parameters for a simple cube, it becomes much more difficult for a complex geometry. Current research is looking to fit instrumentation (e.g. thermal cameras, pyrometers) around the component being built, to monitor the temperature and shape of the melt pool. The data is processed through an algorithm to identify if it is within a target range. If it is outside the range, or approaching the limits, then the algorithm will calculate adjusted values of the laser power and travel speed to correct the behaviour, a process called 'closed-loop control'.

This game is designed to simulate this process for a range of simple geometries, allowing the user to control power and speed in a Selective Laser Melting (SLM) build to achieve a stable melt pool temperature. The game provides a general introduction to metal additive manufacturing and closed-loop control. It demonstrates some of the challenges faced when trying to optimise a manufacturing process, and how researchers are addressing these challenges.

2. INSTALLATION

The game that has been produced is written in Matlab App Designer version 2019a. It is packaged with Matlab Runtime, which is a free-to-download version of Matlab with the ability to run existing scripts, but cannot be used for editing.

If you already have Matlab installed on your computer, installing Matlab Runtime may cause conflicts. In this case, please contact MAPP (mapp@sheffield.ac.uk) for a version of the game packaged without Runtime.

The version currently available is controlled just through a PC keyboard, although a future release will include control using two joysticks run from Arduino. The game is packaged for 64-bit Windows 10 and designed for a screen resolution of 1920x1080. It should still run on screens with different resolutions, but may not display in an optimal arrangement.

To install the game:

1. Download "SLM_Keyboard_1920_1080.exe" from the MAPP website
2. Double click the file and follow Windows instructions for installation
3. The game will automatically download and install the required version of Matlab Runtime (~600MB), unless it is already installed on your computer
4. The game should appear in your start menu

If you have any difficulties installing the game, please contact MAPP (mapp@sheffield.ac.uk).

3. OPERATION

The game has three available component geometries (square, circle, triangle), and three run styles (baseline, closed-loop control, game mode).

The left hand graph shows a plan view of the build proceeding, with the hatches shown as black lines and the colour indicating the temperature of the melt pool (red = hot, blue = cold).

The right hand graph shows a line graph of temperature against build time, with horizontal lines indicating the temperature limits for keyhole melting (boiling) and lack-of-fusion.

At the end of each layer, the game shows a score for the percentage of time spent in the ‘good melting’ zone.

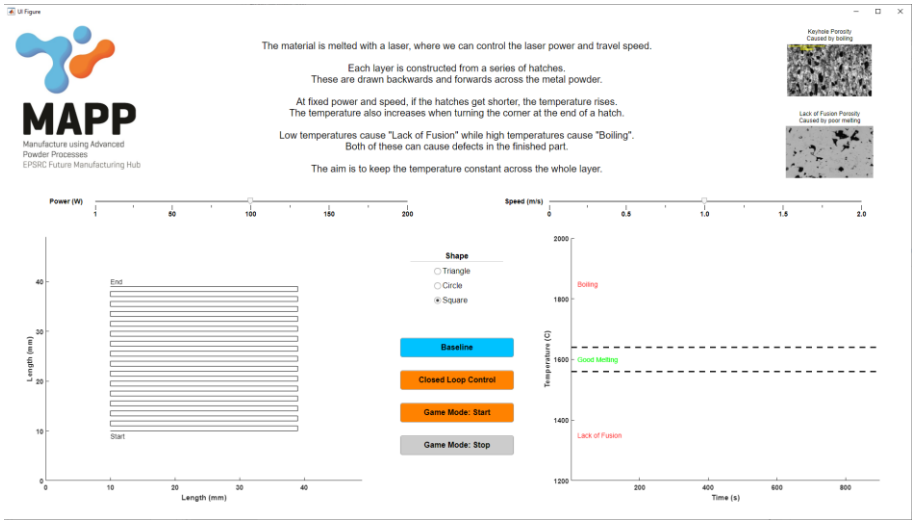

Figure 1: Opening screen, ‘Square’ geometry, no run mode selected

Selecting ‘Baseline’ will run a condition with a fixed power and speed for the whole layer of the selected geometry.

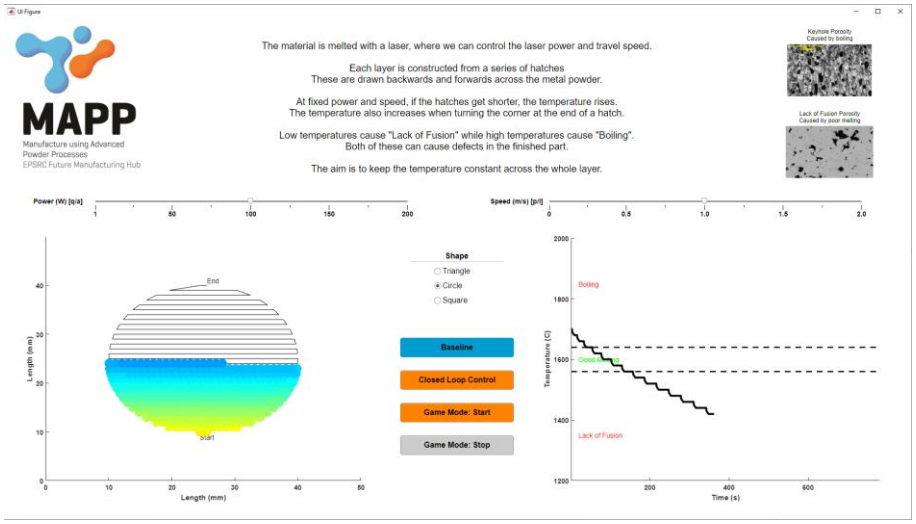

Figure 2: 'Circle' geometry, running in Baseline Mode

Selecting ‘Closed Loop Control’ puts the game into an algorithm-controlled mode.

This uses a fixed speed and power initially, but then adjusts according to the data received back by the computer.

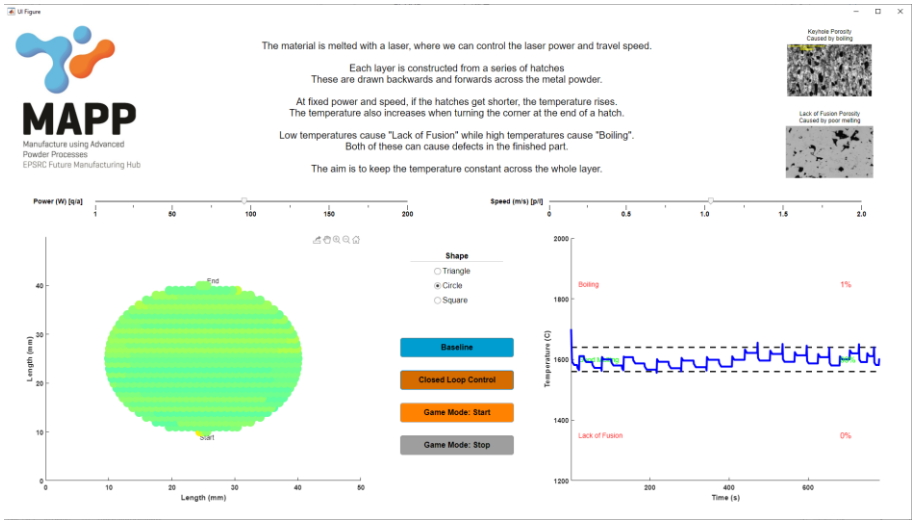

Figure 3: 'Circle' geometry, running in ‘Closed-Loop’ Control Mode

Selecting ‘Game Mode: Start’ puts it into game mode, where the power and speed can be controlled by the user.

If running the keyboard version, the power is controlled by the Q and A keys, while the speed is controlled by the P and L keys.

On pressing ‘Game Mode: Start’ it will count down ‘3… 2… 1…’ and then the build starts.

The user can view the live temperature data on the two graphs, and adjust the power and speed accordingly. The power and speed sliders will adjust to show the current settings.

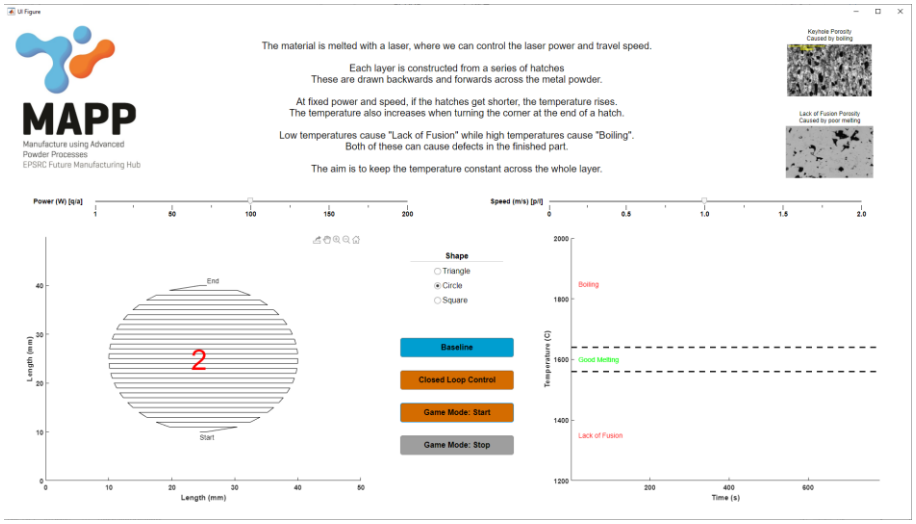

Figure 4: 'Circle' geometry, running in Game Mode during count down

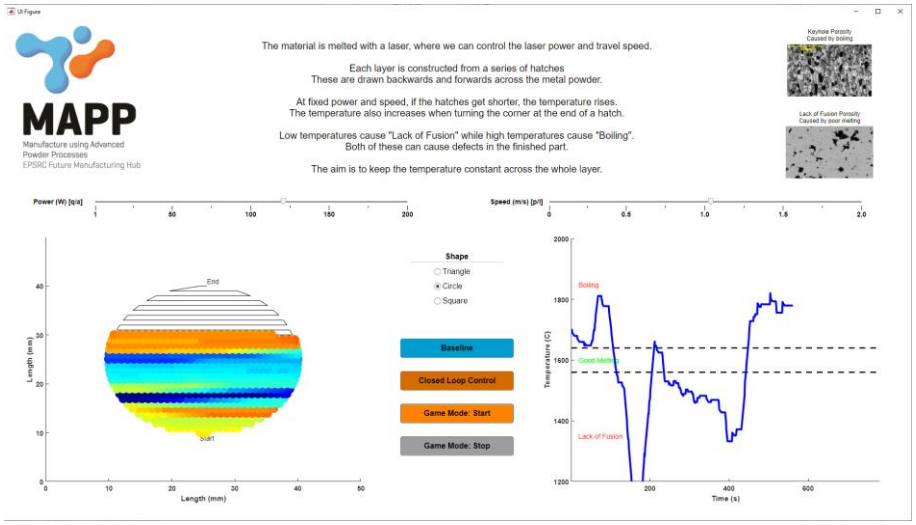

Figure 5: 'Circle' geometry, running in Game Mode with operator control of speed (LH slider) and power (RH slider)

Figure 6: Connections to Arduino

If using Arduino, connect the Arduino to your computer using the USB-B cable.

If the game indicates a connection error, close it, disconnect and reconnect the Arduino and then re-open the game.