Linear Advance

OCTOBER 2, 2017 ~ MATT

In early April, a member of the Prusa i3 Facebook group posted about a feature of Marlin firmware that they had come across named linear advance. Documentation for the algorithm can be found here.

> In a nutshell, **LIN_ADVANCE** enables a pressure control feature which sets the pressure inside the nozzle to the needed one according to the print speed. This way, bleeding edges and rough solid infill can be nearly eliminated.

When your 3D printer extrudes plastic, you'll notice that it takes a little while after you've started extruding for the plastic to actually come out of the nozzle. It also doesn't stop flowing out of the nozzle right after it stops extruding – some still oozes out. This is due to the pressure build up inside the nozzle when the plastic is pushed in. Even though it's melted, it still takes some force to push it all through a tiny 0.4mm hole.

The force needed to move the filament through the nozzle depends on the speed you're trying to push it through. If you push it fast (printing fast), the filament will first be compressed before the pressure inside the nozzle is high enough to extrude the required amount of material.

The documentation also mentions that when the linear advance is properly calibrated, your retraction distance should be able to be reduced to almost zero. This works because by using linear advance, the firmware is able to reduce the pressure inside the nozzle when it reaches the end of an extrusion such that it doesn't keep oozing out. The benefits of linear advance extend far beyond this though. You'll be able to see in the results section how big of an impact it really has.

After some research, I found that this linear advance feature had not yet been integrated in the Prusa branch of Marlin firmware. I added a feature request for it on the Prusa github here and after a few months it was developed and added to the firmware.

Set up

Before you begin, please ensure that you have calibrated your extruder. You need to be feeding the exact right amount of filament for this to work correctly.
Download the most recent firmware from the Prusa drivers page here, and flash it to your printer as usual. This version includes linear advance and a few other great features, especially enabling saving extruder steps to EEPROM again.. yay! There are instructions on how to flash the firmware here if you're not familiar with it.
Now you have the capability to use linear advance on your printer!

Calibration

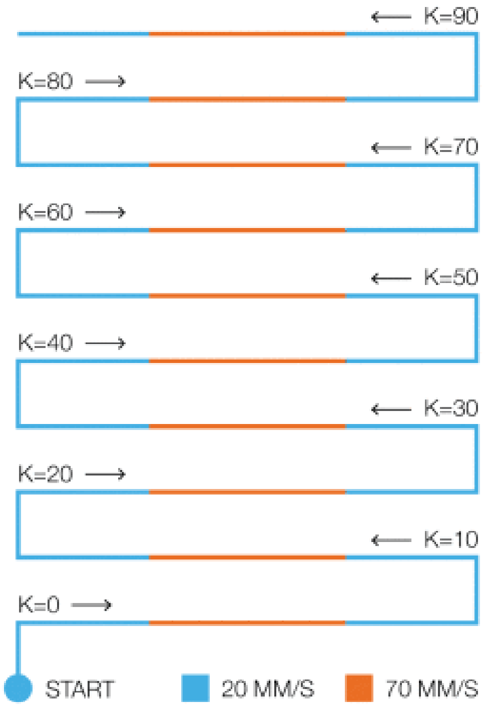
There's only one thing that you'll need to calibrate for the linear advance algorithm, and that is the K factor. This is a measure of how much the printer should compensate for the springy nature of the filament (it compresses and builds up pressure before actually extruding).

A higher K factor means that the filament is more springy, and therefore the printer will compensate for it more. A lower K factor means that the extrusion behaviour is more linear, like the other axes on the printer. A K factor of zero means that there is no compensation at all.

The K factor is heavily dependent on the type of filament that you're printing with, as plastics flow differently when melted and compressed. In order to find the perfect K factor for your filament, you'll need to do a test print that uses different K values in different areas and choose where it looks best. Fortunately, Sebastianv650 on the github thread linked above has created a gcode file that makes this extremely easy. You can download it here: K Factor Test

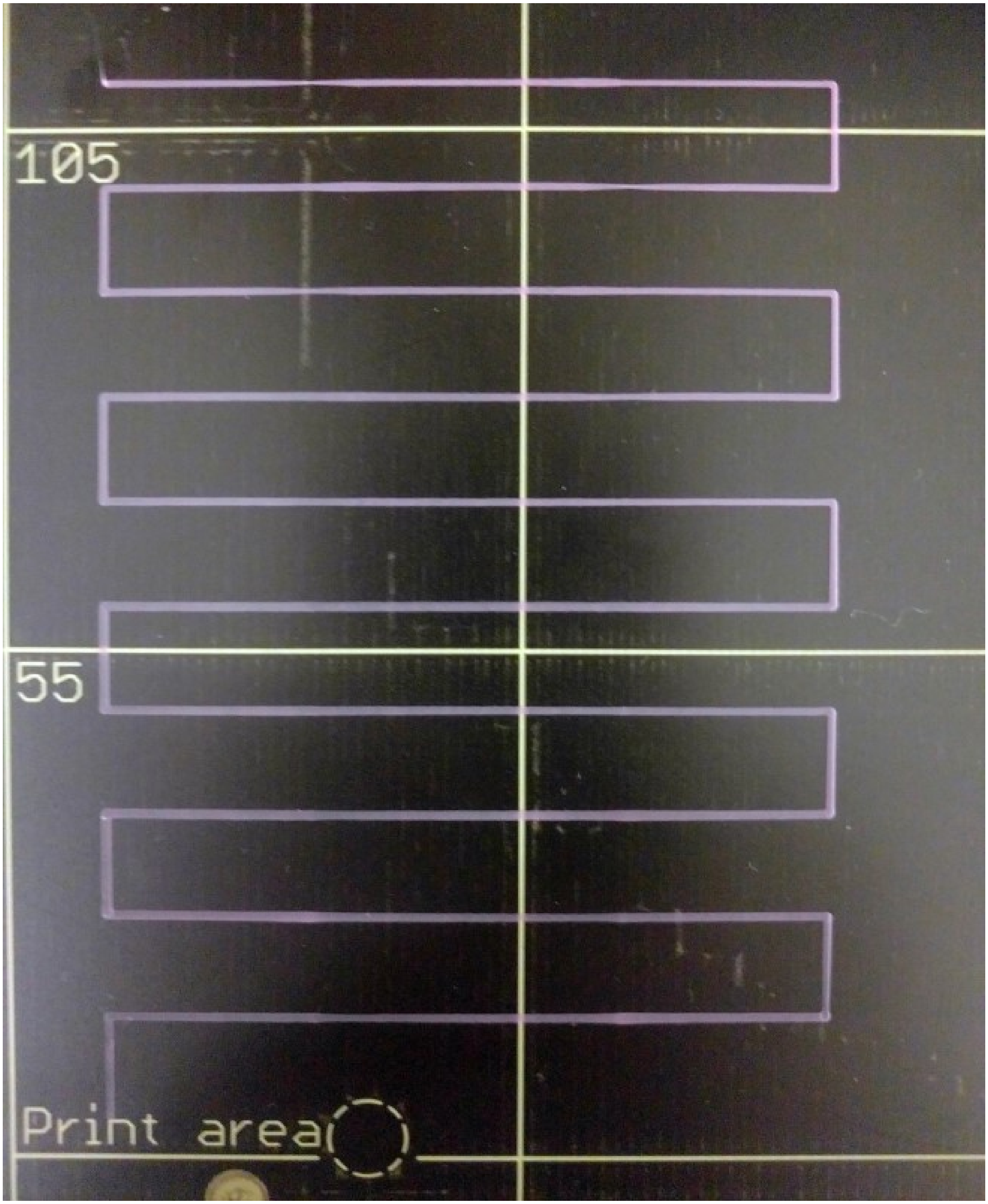
Please note that this gcode uses PLA temperatures and will need to be changed for other material types.

This file will print a zig zag pattern, with each horizontal line using a different K value. The first line is K=0, and each following line increments K by 10. The horizontal lines consist of three segments – it starts slow (20mm/s) for 2cm, then goes fast (70mm/s) for 3cm, then slow again for the last 2cm. This is shown in the diagram below:



With no linear advance (K=0), this would create a thin section of the line where the printer speeds up, and a thicker section of the line where the printer slows down. If the K factor is set too high, then the opposite of this will occur. The line will be thick where the printer speeds up, and thin where the printer slows down. This shows it is over compensating. What we are aiming for is a line of uniform thickness all the way through, as the thin and thick sections represent under and over extrusion.

Once you have printed the K factor test gcode file, observe the lines that are left on the print bed. This is what mine looked like:


105
55
Print area

The picture isn't the best quality, but you can definitely see that the first and last lines have thick and thin areas of under and over extrusion.

Find the line that looks the most consistent and uniform in thickness. In my case, the line for K=40 looks perfect, so that's what I'll set my K value as. If two lines look equally uniform to you, then you can choose the value in between. For example if K=40 and K=50 are both equal, then set your K value to 45.

To set the value, it's as simple as including the command `M900 K##` in your starting gcode, where ## is your K value. The location of the command within the starting gcode isn't important at all. Slic3r PE's new filament specific starting gcode setting is perfect for this, because it allows you to set different K values for different filaments (which you should definitely be doing by the way).
If you download the new drivers from Prusa Research, they will include modified slic3r profiles with higher speeds and some default K values for different materials. I still recommend that you calibrate the K factor for your own filaments, as the values included by Prusa are on the conservative side.

One small note

During development, it was found that some slicers would produce gcode with small areas of very thin extrusion width randomly. With regular printing this isn't an issue, because the fluid nature of molten filament evens out any tiny differences. However, it does cause some issues with the linear advance algorithm and throws it off.

Slic3r does not have this issue, but it appears that Cura does. In order to stop this from throwing off the linear advance algorithm, you can send a command to the printer telling it the layer height, filament diameter, and extrusion width of that particular print.

Do this with the command `M900 W#.## H#.## D#.##` where W is the extrusion width, H is the layer height, and D is the filament diameter. So for a 0.3mm layer height print with an extrusion with of 0.45mm using 1.75mm filament, the command would read `M900 W0.45 H0.3 D1.75`. Enter this into your starting gcode, in addition to the `M900 K##` command.
I would only recommend doing this specific command if you're using Cura, as it will have negative effects if used on a print with variable layer heights or variable extrusion widths.

Results

So what difference does any of this make? Well, here's a list of some differences that you'll notice when using linear advance:

- Lower retraction distance necessary
- No wipe or coast needed
- Seams not as visible
- Sharper corners resulting in greater dimensional accuracy
- No over extrusion at the edges of solid infill
- **Much faster print speeds are possible with no negative effect on quality**

Feel free to crank up your print speeds much more than you usually would. You'll be amazed how the print quality still stays exactly the same. I'm having great results at around 120mm/s, and I could probably go even higher.

As I hinted at above, you should disable coast, wipe, combing, extra restart distance or any other feature like that which is designed to aid with pressure control. You'll also be able to greatly reduce your retraction distance, but that is of course dependent on other things like print temperatures and filament type.

I hope you've found this guide helpful! If you have any problems, please let me know in the comments.