Introduction to
CAD-to-Motion

Nick Repanich
Adjunct Research Professor
Department of Mechanical Engineering and Manufacturing
California State University, Chico
Chico, CA 95929-0789

Table of Contents

INTRODUCTION

More and more people are turning to computers to increase the productivity of their business. In the manufacturing environment, this means CAD/CAM/CIM, or some combination of those. Manufacturing engineers need to cut down development times. They need to produce shorter lead times as a prerequisite just to get the business.

CAD-to-Motion is the process of developing a part or design on a computer, then sending it to a machine. The machine could be as simple as a two-axis stage moving a glue head or burner around, or as complex as a five-axis CNC milling machine. The machine, equipped with motors, will then make the part or draw the design.

This handbook will discuss some of the most common methods, but is by no means exhaustive. It will discuss the different parts necessary, and give some design considerations in each area to make an optimum system. Because each system is different from the next in some respects, this is a collection of general guidelines. Hopefully they will help you to learn a little more about each functional block in a CAD-to Motion system and enable you to make informed decisions when selecting components. Figure 1 shows many of the functions needed.

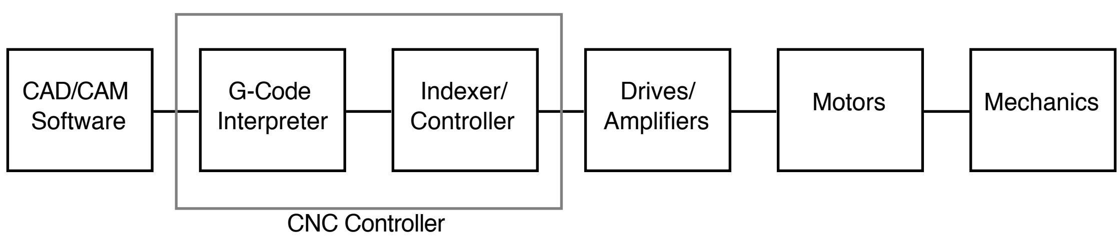

FIGURE 1 - CAD-to-Motion Block Diagram

Not all of the parts shown are needed in every system. For instance, most CNC controllers can take code directly from a CAM software package. The parts which *must* be included are the mechanics, the drives and motors, a controller of some sort, and some way to program the controller.

MECHANICAL CONFIGURATIONS

In this segment, the speeds, accuracies, repeatabilities, torques, inertias, and applied loads will be considered. Many times the first choice to make is whether to move the part, or move the tool head. Some systems use a combination of both. The choice often depends on which piece is more massive. A CNC will usually move the part under the head. The head is much larger than the typical parts made. It is easier to maintain accuracy moving a two pound part than a two hundred pound mill head. On the other hand, if the part is a 2' by 4' foam model of a chair and it is being cut by a light router head, it makes more sense to move the router head over the piece with a gantry-type arrangement.

All CAD-to-Motion systems require a method of converting the rotary motion of a motor into linear motion. The most frequently used method is a leadscrew or ballscrew. Other methods include belt and pulley, rack and pinion, and sometimes linear motors.

There are two main design criteria to match a motor to mechanics. The motor must have sufficient *torque at rated (and sometimes peak) speed* and the mechanics must be stiff enough to achieve stability, and thus the desired accuracy and speed. The combination of these last two factors is commonly called “throughput”, or more technically, the “bandwidth” of the system. The best way to get an idea of the potential for instability is to check the *inertia match* between the motor and the reflected load. Total load inertia is the sum of the inertias of every part which moves in the system. The reflected load inertia takes into account the transmission method between the motor and the system. It is the portion which the motor sees if looking out at the system from the motor shaft.

The less stiff the system, the lower the inertia match should be, approaching perhaps a 0.5:1 ratio. This match is the ratio between the reflected inertia of the load and the inertia of the motor's rotor. As the system gets stiffer, the higher the ratio can be, up to ratios greater than 100:1 with linear motor systems, where the motor essentially becomes the part of the load. As a general rule, the load inertia increases according to $J_{load} = mr^2$ (consult your handy physics book for more inertia formulas).

The first factor to consider in sizing motors is the torque required to accelerate the load to maximum speed. Just reading maximum torques can be misleading. Many times they are listed at zero speed, called static torques. All motors have less torque available at higher speeds than at zero speed.

Assuming a constant acceleration, a system will require the same amount of torque to accelerate at all speeds. Thus, since the least amount of torque is available at high speeds, the motor must be sized at that point. Figure 2 shows this point on a typical trapezoidal move profile.

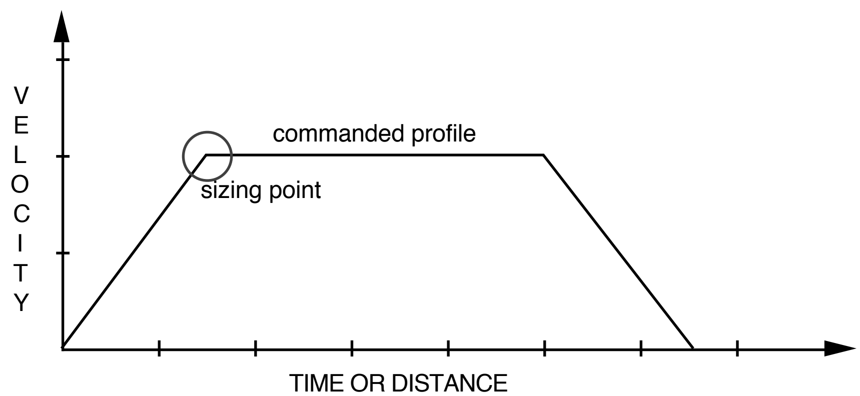

FIGURE 2 - Typical Move Profile

Once the torque and speed requirements are known, the gearing and transmission method can begin to be adjusted to make the whole system more effective. For instance, a common situation is a motor which has enough torque at speed to move the load, but the inertia match is too high for the perceived stiffness of the system, for instance 45:1. There are actually a few choices. The first involves choices of leadscrew.

More physics formulas will be helpful here: Torque required to turn a load is reduced or increased by a gear ratio (R2/R1). The reflected inertia at the motor gets changed by the *square* of the gear ratio $(R2/R1)^2$. The pitch of a leadscrew has the same affects as a gear ratio. For example, a five-pitch screw will reduce the reflected load inertia by 25 and a two-pitch screw will reduce the reflected load inertia by only four. The pitch of the screw does not have any affect on the inertia of the screw itself, since it is directly connected to the shaft.

As the gear ratio (or screw pitch) increases, the speed necessary out of the motor must increase to maintain the same linear speed. So gear ratios are a great way to match speed, torque and inertia. But remember, a motor will only have limited speed available, especially a step motor. There are a couple of cost tradeoffs also. Gear reducers can cost more than the motor, sometimes over a thousand dollars for some of the larger sizes. A larger motor will usually improve an inertia ratio, but larger motors are also more expensive. Most off-the-shelf gear reducers have a maximum input speed of 4000 to 5000 rpm, so even a high-speed servo may only be able to take advantage of gearing to a certain point.

Let's return to a leadscrew moving a linear load and see how to optimize a particular system. The easiest systems have short travels and slow speeds, the most difficult systems to optimize have long travels and high speeds. The most standard leadscrew is a 5 pitch or 0.2 inch lead screw. This means each turn of the screw moves a load 0.2 inches. To go one inch requires five turns of the motor. To get high speed (up to ten in/sec in some CAD/CAM systems), that same five-pitch screw would need to turn up to 50 revolutions per second (rps), or 3000 revolutions per minute (rpm). A stepper will not go this fast with any useable torque, and a servo can double the cost of the motion components. This is not to mention that the critical speed of the screw may be exceeded. A solution in a moment.

The critical screw speed is determined by a couple of factors. First is the length. When a long leadscrew or ballscrew is turning rapidly, it sets up a "jump rope" effect, or a whipping action. The sag in the middle begins to swing around like a jump rope. This will cause torque spikes and will cause the motor to stall. The other factor is the actual rotating speed of the balls in the bearings. There is a speed, sometimes lower than the "jump rope" or critical speed at which the balls in the bearings begin to slide instead of roll. The system will work, but later it will be discovered that the ball bearings have been destroyed, so a safety margin is in order.

To get the combination of high speed and long travel, first the pitch should be decreased, probably to 2 pitch. The immediate result is the motor and screw only have to turn at 20 rps, but some accuracy is given up. The diameter of the screw would then need to be increased, so it does not sag enough to whip at the speeds desired. These steps increase the torque required by 5/2, and the reflected load inertia by 25/4. Increasing the diameter also increases the inertia of the screw itself by a factor of the radius squared, and the cost of the screw significantly. Usually all of this forces a larger motor, both to get the higher torque required to achieve the same accelerations, and to keep a reasonable inertia match. Sometimes, it can mean changing to a servo instead of less expensive stepper. Speed, length, and accuracy just don't all fit in the same system very well.

So in general, leadscrews and ballscrews work best when more resolution and accuracy are needed, and when the linear speeds required are low (1-10 inches per second (ips)). They are also a good method if there are significant weights or forces involved such as moving a mill head through metal.

The next two mechanical configurations are the belt and pulley, and the rack and pinion. These usually offer the highest speeds, longest travels and lowest accuracies. These configurations work best in applications which do not require a high pushing force, such as glue dispensing.

One of the important design factors in a belt and pulley configuration is the reflected load inertia. With a pulley attached to the motor and one on the other end, perhaps an idler and the belt, the reflected inertia gradually builds with each additional load component. If the pulley diameter is over a few inches, the 1:1 inertia ratio will be reached with just the drive pulley, without even considering the load. The load inertia gets amplified by the square of the radius of the drive pulley. The best action to take is to keep the drive pulley smaller, and the motor larger (higher inertia rotor) and faster, to compensate for the smaller pulley. Essentially this becomes an implicit gear ratio, which has already been covered. Belts and pulleys work best when moving light loads with limited resolution and accuracy.

There are a couple types of belts that are most common. For more accurate applications, a timing belt is a good choice, though the available lengths are sometimes a limiting factor. Cables or cable chains are good for longer travels, but accuracy will suffer because the tooth fit around the pulley is not as tight. If the application requires high force which would break a cable or belt, a regular chain must be used, but when this becomes the case, it may be better to go to a rack and pinion.

Rack and pinion systems are often used when the load is too heavy to use a belt, or when the length of travel is long and a belt is impractical. The tradeoff from a belt and pulley is that now the motor moves with the load, adding to load inertia. In a light load system (less than ten pounds) this could mean most of the power in the motor is used to move itself. In heavier systems, it is not such a large factor. Support of the motor can be an important factor here, for high accelerations can exceed the radial load rating of the motor bearings, and destroy the motor. An extra bearing structure has to be provided, adding to the cost.

LINEAR MOTORS

Somewhere in the middle between mechanics and the actual motors themselves, lies the linear motor. This may be one of the most efficient ways to achieve high speeds and not have a lot of extra expense in bearings, screws, belts, etc.

The linear motor has much in common with rack and pinion systems, where the load is attached directly to the motor. There are two parts to the system, a platen (stays in one place) and a forcer (the motor). The forcer floats over the platen via some type of bearing. Most systems will use some sort of linear guide, with higher end systems using an air bearing. With the air bearing, there are no wearing parts, and no limitations in speed or acceleration.

Linear step motors can achieve speeds up to 75 ips, and accuracies of about 0.0035". The drives can give the motor a resolution of around 10,000 steps per inch. Linear servo motors can achieve speeds up to 100+ ips, and accuracies as good as the feedback system will allow. Platen lengths up to 12 feet can be achieved, and still maintain the same accuracy and speed. They are mechanically simple. For an X-Y gantry, the load is mounted to one forcer, and then its platen is mounted to the forcer of the other axis at a 90 degree angle.

Of course, there are some critical application considerations. The environment around the forcer is important. Dry dirt or dust is fine with the air-bearing forcers, as it gets blown off the platen ahead of the forcer. With roller bearings, dirt is a big problem. With either type, paint, glue, chips or anything likely to stick to the platen can wreak havoc. Remember, with a linear motor, the inside of the motor is the 0.0005" air gap between the forcer and the platen, and is open to the elements. To overcome this, forcers can be covered with bellows, run upside down, or surrounded by shrouds. The other main application consideration is mounting the load. Overhanging loads, heavy loads, or any type of load which tends to twist the forcer will create problems. For this reason, most designers will put a separate linear bearing structure on the load, and just use the linear motor to push and pull.

DRIVES AND MOTORS, STEPPERS VS. SERVOS

There are a few basic differences between steppers and servos. A servo *system* (not motor) has continuous feedback from a resolver, tachometer, or encoder, allowing the driver (or amplifier) to send current to the motor only when there is a difference between the commanded position and/or velocity, and the actual position and/or velocity. Thus, a servo runs cool, and the shaft is not held in a stiff or locked position. A step motor system has full current running through the motor at all times. The rotor is very stiff at zero speeds, and only exceeding its torque capabilities will cause it to be moved when not commanded.

A servo will run much faster than a step motor system. Speed in a motor is generally limited by the difference between the power supply (rail) voltage and the back-EMF voltage. This back-EMF voltage is due to eddy currents in the motor windings. These eddy currents pass through the resistance of the copper windings, and create the back-EMF voltage. This counteracts the power supply voltage and effectively reduces the applied voltage to the motor. Since the winding resistance always remains constant, by Ohm's law, the current gets reduced. Torque is proportional to current, thus less torque is available at the higher speeds.

A step motor has 50 poles, compared to 4 or 8 in a typical servo motor. This high pole count of the stepper motor drastically increases the amount of eddy currents created. Thus, the stepper motor torque will fall off much more rapidly at higher speeds, especially above 10-15 rps. The servo will be able to achieve much higher speeds, up to 100 rps. When the motor is hooked up to a leadscrew, this means a step motor will move a typical load up to a maximum of about 4 ips before it stalls. A servo will go from there on up to 15 ips or more, usually on a ballscrew. Figure 3 shows typical speed/torque curves for stepper and servo systems.

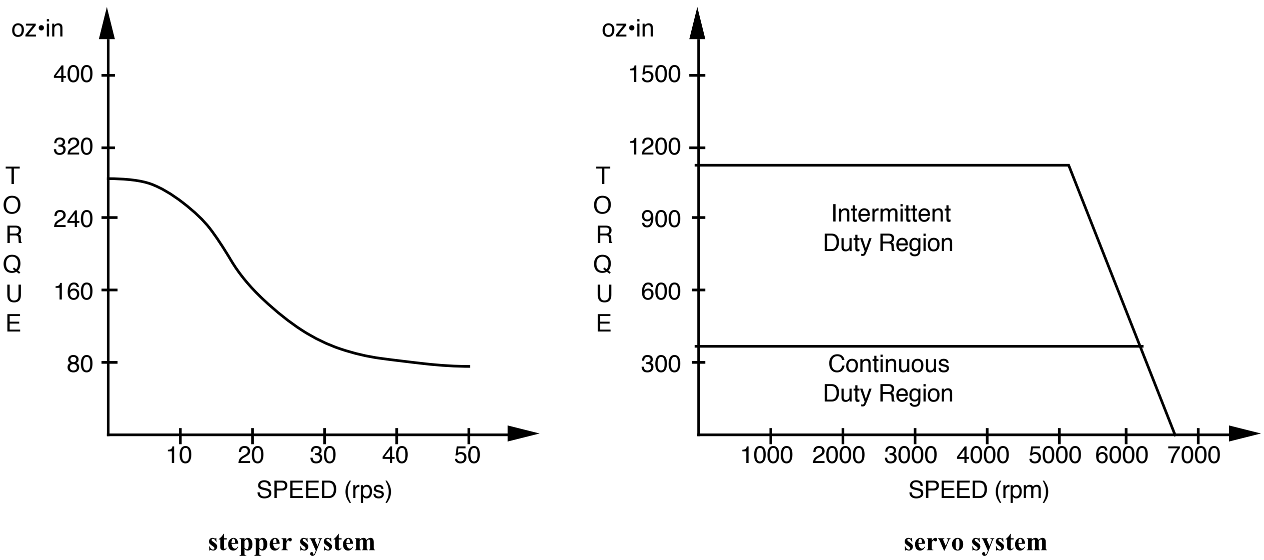

FIGURE 3 - Typical Speed/Torque Curves

CONTROLLERS

The purpose of the controller is to command the drives and motors how far and how fast to travel. There are two general types of controllers used in CAD-to-Motion applications. The most common in machine tool applications is the CNC controller. This is what would normally come on a machine tool, if it came with controls and motors. They usually have their own programming screens and use G-codes, which is the accepted standard for CAD-to-Motion programming.

The other type is the more general multi-axis motion controller. It is designed as a flexible controller which can handle multiple axes in many kinds of applications, of which CAD-to-Motion is just one. They usually have their own language and if they accept G-code, there is usually a converter somewhere in the software. To easily program and make parts or designs in a CAD-to-Motion application with these general controllers, separate CAD/CAM software packages are needed. The big advantage to this type of system is cost. If there is already a computer available, it may be up to $5,000 for the controls, instead of $15,000. The limiting factor in this type of system is generally the internal conversion from G-code to the controller's own language.

PROGRAMMABILITY

Sometimes programmability can be an important part of the decision on what type of controls to buy. How much time is spent programming, and how much making parts? What happens to lead times if a lot of time is spent programming, and how much does it really cost in lost business? Does the company already have a CAD program?

To help answer some of these questions, consideration should be given to the type of parts being made and whether they need to be made using tool offsets. If the part or design can be made using just centerline movements, a full-blown CAM software package may not be needed. Applications which fit this are sewing, laying down glue, low accuracy cutting, engraving and drawing. Perhaps all that is needed is a controller which can be programmed directly in G-code, or even its own language. This is not considered CAD-to-Motion because there is no CAD involved. If the controller does take G-code, the part may then be drawn in CAD, and the CAD program's G-code output may be used, if it has one. The only problem here is matching the two codes, since not all G-codes are standard for every controller. If the CAD package does not output G-code, converters are available which convert a simple .DXF file (most all CAD programs create this) to G-code.

If tool offsets are required to make the correct size parts, if will be hard to use a regular CAD program, because of the difficulty in adding tool offsets. It can be done, and some CAD programs even have offset capability, but they still won't be designed to have a tool path look ahead, or tool changing functions. Without a CAM program, the responsibility of compensating for tool width and tool path fall on the user, and how well they can visualize what the tool is actually going to do. This is somewhat of a trial and error approach, with hopefully less error. If expensive parts are being made, the extra cost of a proper CAD/CAM program ($5K to $15K) will be money well spent.

Some CAD-to-Motion systems only need two axes of control, such as flat glue dispensing. Others need two and one half axes, meaning the Z axis moves up and down in a linear motion, sometimes timed to the X and Y axes. The next level of sophistication in programming is true 3-D, which means parts can be made where spheres will melt together, creating a smooth surface. As the machining becomes more complicated, and the number of axes increases to four and five, so does the cost of the machines and the programming software, and more than just proportionally. The limitation to what you can do with CAD-to-Motion is determined by the hardware and software, and a budget to make it happen.

If this handbook has not gone into enough depth on any one subject, it is mainly because there tends to be many possible configurations to consider in every application. We would be happy to discuss any of these topics in further depth.