Design Fundamentals

Weight Classes

The lightest combat robots ever built have less than 35 grams (35g), but they are so rare that there is no name yet for this weight class. Fleaweights (a.k.a. nanoweights or UK fairyweights), in general with a weight limit of 75g (or 50g depending on the event organizers), are also very rare. Fairyweights (up to 150g, known as antweights in the UK) are becoming popular, however there are still few events including them. Antweights (1lb) and beetleweights (3lb) are the most competitive among the “insect” classes (ants, beetles, fleas...). There are also autonomous ant and beetle classes.

The kilobots (1kg) events only exist in Canada, and the 15lb class is only for students between 12 and 18 years old, who participate in the competition BattleBots IQ. The Mantis (6lb) weight class has not really caught yet, there are very few robots in it. Featherweights (30lb) are becoming increasingly popular, especially in Brazil.

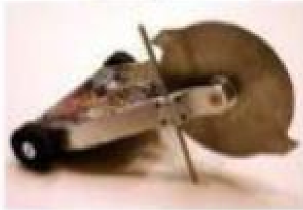
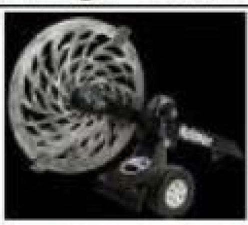
The 12lb and 30lb Sportsman’s classes are special categories where all robots must have an active weapon, wedges of any form are forbidden, and spinners are severely restricted.

Possibly the most competitive classes are the hobbyweight (12lb), lightweight (60lb) and middleweight (120lb). Heavyweight (220lb) is the most famous class, in spite of having nowadays much fewer competitors than when BattleBots was televised.

Unfortunately, super-heavyweights (up to 340lb, or 320lb in UK events) are in decline, their apogee was also during the BattleBots era. The heaviest class is the Mechwars megaweight (390lb), exclusive to the Twin Cities Mechwars competition, however few robots exist.

There are still heavier robots, such as the MonsterBots, however events involving them are very rare due to logistic problems and high costs involved.

Back in 2006, when the first version of this tutorial was released, most Brazilian combat robots were middleweights (but not anymore, since the hobbyweight and featherweight classes started in Brazil). Because of that, several examples in this tutorial make reference to middleweights. However, the contents of this tutorial can be applied to any robot size, as it will be discussed in the next section, which deals with scale factor.

 *class still without a name - 35g*	 *Fleaweight - 75g*	 *Fairyweight - 150g*
 *Antweight - 1lb (454g)*	 *Kilobot (Canada) - 1kg*	 *Beetleweight - 3lb (1.36kg)*
 *Mantis - 6lb (2.72kg)*	 *Hobbyweight - 12lb (5.44kg)*	 *BBIQ - 15lb (6.80kg)*
 *Featherweight - 30lb (13.6kg)*	 *Lightweight - 60lb (27.2kg)*	 *Middleweight - 120lb (54.4kg)*
 *Heavyweight - 220lb (99.8kg)*	 *Super-Heavyweight - 340lb (154.2kg)*	 *Mechwars Megaweight - 390lb (176.9kg)*

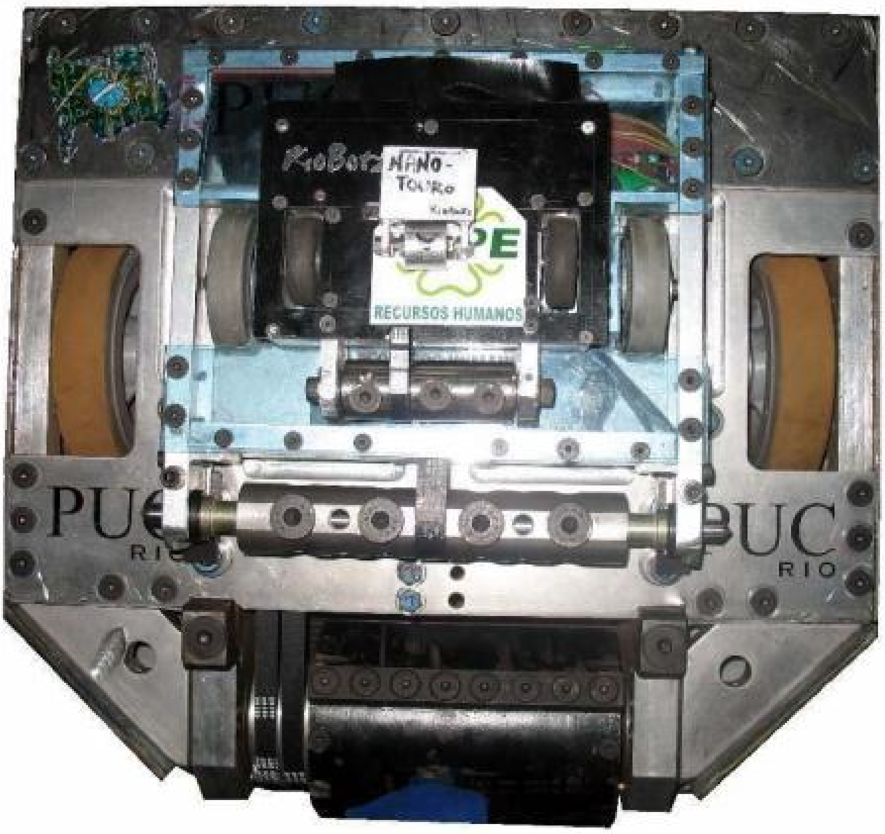
Scale Factor

One important thing to keep in mind during the design phase of a combot is the scale factor. If you grew up in all your body dimensions, you would be twice as tall, and with eight times your weight (because your volume would be multiplied by $2^3 = 8$).

However, the area of the cross section of your bones and muscles would only have been multiplied by $2^2 = 4$. Since the cross section area (of a column of a building, for instance) dictates the resistance and load capacity, you would be 8 times heavier but only 4 times stronger. Conclusion: the larger the scale, the worse the force/weight ratio.

To compensate for that, your bones would have to be proportionally wider and shorter so that they wouldn't fracture or buckle. This is why rhinos and elephants have such wide and short legs. On the other hand, when reducing the scale, the inverse effect happens. An ant is about 100 times smaller than a human being, and because of that its weight is about 100^3 times smaller, however its force is only 100^2 smaller. As a result, ants can carry objects $100^3/100^2 = 100$ times heavier (relatively) than a human being would be able to. That estimate is confirmed in practice: a typical human can carry an object with half of his/her weight, while it was already proven that ants can lift loads 50 times their own weight, a factor of 100 more!

You should be wondering: what does this have to do with combots? Everything. If for instance you have designed a hobbyweight that is resistant and works well, you could take advantage of a lot of its design to build a middleweight, as long as you keep in mind the scale effect. To do that, you would need to multiply the weight by 10, which happens when you multiply all the robot dimensions by the cube root of 10, which results in a scale factor of 2.15.



The picture to the right shows a few drumbots, the middleweight *Touro*, the hobbyweight *Tourinho*, the beetleweight *Mini-Touro*, and the mock-up of a fleaweight *Pocket-Touro*. The scale factor between the 12lb *Tourinho* and the 120lb *Touro* is a little lower than 2 (which is close to the theoretical 2.15, but this suggests that *Tourinho* could still have been optimized to arrive in that 2.15 value, since both robots have similar shapes and weapons). This rule works very well in all scales, as long as the robots are similar: *Touro* is 40 times heavier than *Mini-Touro*, and the scale factor measured among them is about 3.25, very close to 3.42, the cube root of 40!

The question is: following the reasoning of the ant and the human, is it true that a middleweight such as *Touro* is, relatively, about $2.15^3/2.15^2 = 2.15$ times less strong, agile, powerful and resistant than the hobbyweight *Tourinho*? Yes and no. *Touro* will probably be relatively less strong and agile. If for instance *Tourinho* used a pneumatic cylinder, which has a force that depends on the piston area, a cylinder scaled to 2.15 in *Touro* would be only 2.15^2 times stronger, while the robot would be 2.15^3 times heavier. The drive system accelerations, which depend on the ratio between the robot's traction force and mass, would be compromised as well. This is why, comparatively to their sizes, the insect robots seem to be much more agile.

However, *Touro* won't be relatively 2.15 times less powerful and resistant. In the case of a pneumatic cylinder, its energy comes from its internal volume (multiplied by the operating pressure). Therefore, a cylinder scaled to *Touro* would have 2.15^3 times more volume and energy, which is compatible with a weight increase of 2.15^3 times. The same is observed, for instance, in electrical direct current (DC) motors. In practice, the power/weight ratio of the best DC motors does not depend much on the scale factor. Otherwise, it would be worthwhile to replace a large motor with hundreds of small ones in parallel. Since power generates energy, and energy generates damage, *Touro* and *Tourinho* would have the same relative power and therefore the same relative damaging capabilities.

This conclusion is not very intuitive, especially when you consider that both *Touro* and *Tourinho* are able to fling opponents from their same weight classes up to the same 3 feet in the air. One can think that *Tourinho* would generate more destruction, because the relative throw height would be larger if compared to the robot size. But that same height is not surprising, it is verified by the expression of the potential energy $E = m \cdot g \cdot h$, where m is the robot mass, g is the acceleration of gravity, and h is the height reached in the throw. As the E/m ratios of *Touro* and *Tourinho* are approximately the same (as discussed before) and g is a constant, the height h should be the same. Although small robots are flung to a greater height with respect to their size, both energy and resistance depend on the cube of the scale factor. Therefore the destruction power (damaging capability) is relatively the same.

But why are *Touro* and *Tourinho* equally resistant, considering that the resistance of a column depends on the square of its scale and not on the cube? In fact, if *Touro* used in some way slender columns, subject to compression and buckling, it would be relatively 2.15 less resistant than *Tourinho*, following the "ant reasoning" and the dependence on the square of the scale. But the best combots are compact and robust, without slender parts. The most important loads that act in their compact structure are due to bending and torsion. But the resistances to bending and torsion depend on the cube of the scale factor (a shaft with diameter d, for instance, has bending and torsion resistances proportional to d^3), not on the square such as in buckling. Therefore, the bending and torsion resistance-to-weight ratios are still similar for both *Touro* and *Tourinho*.

The conclusion is that the scale factor can be used directly in the entire robot, without any significant loss of the power-to-weight or resistance-to-weight ratios. For instance, if you multiply by 2 the robot size, its weight is multiplied by 8. The analogy with ants would say that the diameter of a shaft in this robot would need to be multiplied by the square root of 8, about 2.83, to maintain the same resistance-to-weight ratio. That would be necessary if you were designing the column of a building, subject to buckling, but this is not the case for combots. In that case, it would be enough to multiply by 2 the shaft diameter to keep the same resistance-to-weight ratio. This is useful for two reasons: first, this means that you can apply the same scale factor (2, in this case) to all the individual components of the robot; and second, you save weight, because the shaft with diameter multiplied by 2.83 would have twice the weight of the one multiplied by 2.

But there is another factor to consider: shafts in combat robots are usually relatively short, which are subject to high shear stresses. In addition, great impacts can generate tensile stresses or significant compression. The resistance to those traction, compression and shear stresses in a shaft with diameter d is proportional to d^2, not to d^3, taking us back to the ant analogy. As during combat we cannot predict which stresses will be more or less significant, and as the shafts are very critical components that cannot break or get bent, it is desirable to be conservative and adopt the higher factor $2^{1.5} = 2.83$ for the shaft from the previous example.

In summary, you should use the scale factor to multiply (or divide) the dimensions of all the robot components, except for the most critical ones such as shafts, where the scale factor should be raised to the power of 1.5. Don't use such larger factor in the entire robot, otherwise the robot might gain too much weight (when upsizing it) or lose strength (when downsizing it). Use the higher factor only for multiplying shaft diameters or for the dimensions of a few other critical components.

All those considerations are not just philosophical, they are verified in practice. Steel shafts used to drive the wheels of several combat robots typically have, in average, a diameter of about:

- 13mm (about 0.5”) for lightweights (60lb);
- 18mm (a little less than 0.75”) for middleweights (120lb);
- 25mm (about 1”) for heavyweights (220lb); and
- 31mm (a little less than 1.25”) for super-heavyweights (340lb).

Comparing lightweights and middleweights with similar aspect, the theoretical scale factor would be $(120lb/60lb)^{1/3} = 2^{1/3} = 1.26$, and the ratio between the shaft diameters is 18mm/13mm = 1.38, a value incredibly close to $1.26^{1.5} = 1.41$.

Between middleweights and heavyweights, the theoretical scale factor is $(220lb/120lb)^{1/3} = 1.22$, and the diameter ratio is 25mm/18mm = 1.39, very close to $1.22^{1.5} = 1.35$.

And between heavyweights and super-heavyweights, the theoretical factor is $(340lb/220lb)^{1/3} = 1.16$, and the diameter ratio is 31mm/25mm = 1.24, which also agrees extremely well with $1.16^{1.5} = 1.25$.

The bottom line is that theory, combined with common sense, is a very powerful design tool in practice. Imagine how many shafts have been broken in combats worldwide before arriving at these optimized diameters, while with a few simple calculations we've arrived at the same result.

Note however that these are average diameters, the actual values may vary depending on the steel alloy used in the shaft, number of wheels and combat robot type. The combat robot types are discussed next.

Combat Robot Types

After choosing the weight class of your robot, the next step is the choice of the robot type. There are several types of combat robots. None of them is the best. It is a rock-paper-scissors game. Or, as combot builders say, a wedge-spinner-hammer game. The wedges tend to flip over the spinners, which in turn tend to cut off hammers, which tend to puncture or damage the wedges. But they only tend to.

The truth is that a well designed robot can win against a robot of any type, independently of the trends. In the figure below there is a diagram showing such trends for several types of robots. In the figure, each robot has a tendency to win against the one it is pointing to. But a good design and a good driver can completely change this.

There are basically 16 types of combots: rammers, wedges, lifters, launchers, thwackbots, overhead thwackbots, spearbots, horizontal spinners, sawbots, vertical spinners, drumbots, hammerbots, clampers, crushers, flamethrowers and multibots, which will be described next.

Other types of robots exist, but they can almost always be categorized into one of the 16 types above, or in a combination of them. Consider for instance the robot known as the "Swiss army knife," one with two or more weapons. The Swiss army knives in general are not very efficient, it is better to concentrate the weight on a single powerful and efficient weapon than on two or more smaller weapons. It may be a good idea when the weapons act together, at the same time against an opponent. For instance, the 2006 version of our middleweight spinner Titan (pictured to the right) used a wedge (the weapon of the wedge robots) together with its blade to lift lower opponents and hit them.

Most secondary weapons that are efficient in practice are wedges, used for instance to slow down the spinning bar of an opponent before it is safe to attack it with the main weapon.

There is also the "chameleon" robot, with weapons that can be switched during each pitstop depending on the opponent from the next fight. These robots can change their type very quickly, taking advantage of the best each type has to offer. The superheavyweight Shovelhead (pictured to the right) has 15 different weapons that can be installed on its articulated front, one for each type of opponent.

A few accessories can make a big difference. For instance, it is not a bad idea to install some sort of bumper if you'll face a spinner. There are even specific accessories against specific robots, such as using a long stick to hold the shell spinner Megabyte by its vertical tube, as pictured to the right, to repeatedly shove it against the arena walls. However, it is not easy to design efficient weapons that can be quickly dismounted and assembled during a pitstop.

The 16 main types of robots are discussed next. Several photos below were taken from the BattleBots website, www.battlebots.com.

Rammers

Rammers are ramming robots, they damage the opponent throwing themselves against them or pushing them against the borders of the arena. They usually have 4 (or more) wheel drive, wide wheels with high traction, a sturdy drive system, robust armor, high resistance to impacts, and they don’t have weapons except for their passive shields. In general, they are invertible (they can be driven upside down). They need to be capable to push at least 2 times their own weight. They are effective against robots with spinning weapons, such as spinners, drumbots and sawbots.

Wedges

Wedges are robots with a sloped plate shaped as a wedge. They usually have 2 or 4 wheels, with a very resistant drive system. They can be invertible or not. Despite rarely causing damage directly, they are a good tactic against spinners, making them flip over when hitting the wedge. Wedges win against their opponents by entering underneath them and dragging them around the arena, or flipping them at high speeds. Fast wedges usually reach 20 to 25km/h (12.4 to 15.5 mph). The front of the wedge should not be made out of sheet metal, because it can get easily bent and lose functionality.

Use thick plates chamfered at the edge to withstand the opponents’ impacts. Wedges are good against rammers and robots with spinning weapons, and they are vulnerable mainly to other lower, faster and more powerful wedges.

Lifters

Lifters are robots capable of lifting the opponent, immobilizing it or turning it upside down. They are efficient against robots that depend on traction such as rammers and wedges, or robots that have protuberating parts that can be reached by the lifting arm. They are inefficient against thwackbots and overhead thwackbots, because they are difficult to catch and they can work inverted. Lifters are vulnerable to spinners. The lifter design involves a slow linear actuator to lift the opponent, which can stop in the middle of its course. In this way, one can lift an opponent and drag it around the arena instead of just flipping it over. A few lifters use pneumatic systems, but most of them use electric motors with linear actuators. Place the batteries as far behind as possible in the robot, to act as a counterweight when lifting an opponent. The front wheels need to have high torque and high traction, because the robot weight will move forward when lifting and dragging the opponent. A few robots, such as the famous Sewer Snake, use active wedges that also work as lifters.

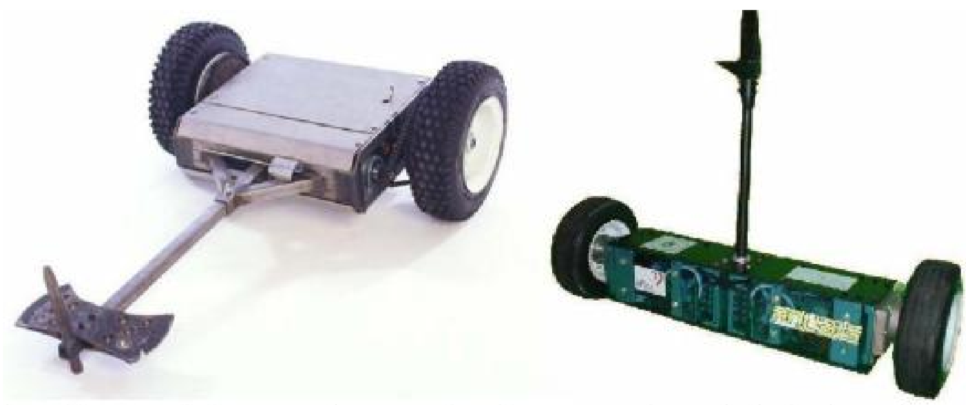
Launchers / Flippers

Launchers (or *flippers*) are lifters on steroids, being capable of flinging the opponent high into the air. The opponent not only can be flipped over, but it can also suffer great damage when hitting the ground. Therefore, launchers are good against opponents with weak chassis, or batteries and electronics without protection against impacts. Launchers need pneumatic components with large diameters actuated by high pressure air or CO_2. Eliminate all needle valves from the system, or use a big accumulator, to guarantee the high gas flow necessary to power the weapon.

Thwackbots

Thwackbots are usually 2-wheeled robots, invertible, which rotate all their structure in the same place at high speeds. They usually have one or more long rods with a hammer, axe, or some piercing weapon. They use the energy of their own drive motors to power the weapon, leaving more weight for their armor. The tires need to be narrow, otherwise they will suffer large friction losses when trying to turn on a dime at high speeds. The wheels, besides being narrow, cannot be too far apart. The closer they are, the faster will be the final angular speed of the robot, however the slower will be the acceleration and the harder will be to drive on a straight line if necessary. The drive motors need to have high RPM. The main problem is that most thwackbots are not capable of moving around to pursue their opponent while they are spinning. Very few thwackbots have developed successful mechanical or electronic systems with that purpose, as studied in chapter 6. Thwackbots are sometimes called full-body spinners, for obvious reasons.

Overhead Thwackbots



Overhead thwackbots use their weapon in an overhead movement, instead of a horizontal one such as with the thwackbots. They have 2 wheels and a long rod, which rotates when the drive motors are reversed, attacking the opponent’s top. It is important that the motors have high torque, because the weapon has only a 180 degree course to acquire its maximum speed. Unlike thwackbots, the wheels should be far apart to help it move on a straight line and to increase the precision of the attack. The tires should be wide to maximize traction. The center of mass of the robot needs to be very close to the line passing through the axes of the wheels, to guarantee that it can lift the weapon to attack.

Spearbots

Spearbots have a long and thin penetrating weapon, usually pneumatically actuated, which tries to penetrate into the walls of the opponent's armor and damage vital internal components. The weapon needs to be resistant and sharp, reaching the largest possible speed. Some conicity in the spear tip is a must to avoid it getting stuck in the opponents. They usually have 6 wheels, to guarantee high traction, necessary so that the robot doesn't move too much backwards during the attack. They are not too efficient, except against robots with thin lateral armors or with exposed vital components. A few robots tried to implement attacks with tethered projectiles (projectiles are forbidden unless they are tethered), but they ended up converging to the spearbot design.

Horizontal Spinners


Horizontal spinners are the most destructive robots. They have a bar, disk, shell or ring that spins at high speeds. When the weapon spins very low near the ground, the spinner is called an undercutter. Ring or shell spinners (such as the robot Megabyte) spin their entire ring or shell-shaped armor, being capable of storing a high kinetic energy, becoming almost impossible for the opponents to reach them without being hit by their weapon. The weapon needs to spin as fast as possible, and you should be able to accelerate to a speed that can cause significant damage in less than 4 seconds. Spinners that take longer than 8 seconds to accelerate may never have a chance to damage a resistant and aggressive opponent. Spinners need to be fast to escape from their opponents while they spin up. Their greatest disadvantage is that, in general, they are not invertible, depending on luck to flip back. To compensate for that, a few robots such as The Mortician and Last Rites, called offset spinners, have moved their blade forward, making them invertible. However, by doing so the robot ends up with large dimensions, compromising its robustness, its back is vulnerable to attacks, and its center of gravity is moved too much forward, away from its wheels, decreasing traction.

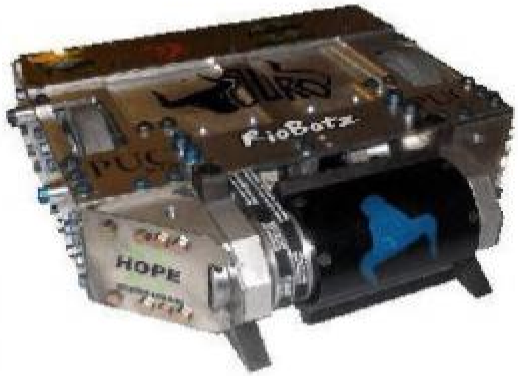
Sawbots


Sawbots have abrasive or toothed disks powered at high speeds by a powerful motor. They are in general combined with other designs, such as wedge-saws. The saws have little efficiency to cut through the opponent, especially if it is trying to escape. They can easily cut sheet metal and Lexan, but they can hardly cut any metal plates during a fight. Their greatest advantage is the cosmetic damage they cause, generating a shower of sparks, scratches and shallow cuts, which can impress judges and guarantee victory in a close match. Saws that rotate in such a way to lift the other robots have high risk of getting stuck on the opponent, breaking or bending. Saws that rotate downwards reduce this problem, however they increase the chance of self-flipping over.

Vertical Spinners


Vertical spinners are sawbots on steroids. Unlike sawbots, in general they use large diameter disks with very few teeth, or bars, spinning on a vertical plane. Damage is caused by both impacts: when the opponent is hit by the weapon and thrown into the air, and when it hits the ground. Vertical spinners need to have a wide base so that they don't tumble when turning due to the gyroscopic effect of the weapon (discussed in chapter 6). The impact force is transmitted to the ground, and not sideways such as with spinners, allowing them not to be flung to the sides due to their own impact. Their disadvantages are having their lateral and back exposed, and having a hard time making quick turns due to the gyroscopic effect. They have problems against very low wedges and tough rammers. The fights against horizontal spinners are extremely violent and fast, and they can go either way, although vertical disks with large diameter usually lose to powerful horizontal bars.

Drumbots

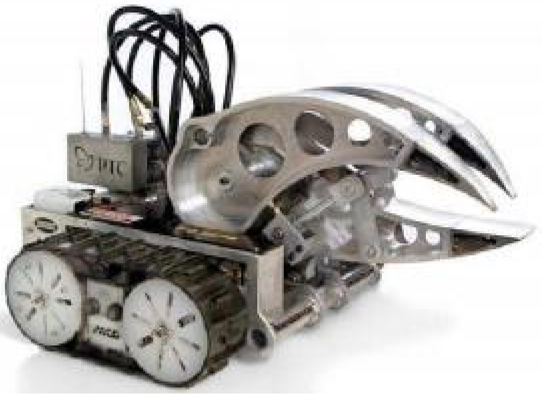


Drumbots have a spinning drum or eggbeater with teeth, in general powered by belts or chains, horizontally mounted in front of the robot. They usually rotate in such a way to launch the opponent, turning it over or causing damage from the impact with the weapon or with the ground. Drumbots are more compact versions of vertical spinners, with less moment of inertia in the weapon. This allows a shorter acceleration time for the drum, however causing less damage to the opponent. They are very stable due to their low center of gravity, they can be invertible, and they make turns more easily than vertical spinners due to the smaller gyroscopic effect (discussed in chapter 6). Wider drums allow drumbots to reach their opponent without needing a perfect alignment. The acceleration time of the drum should be at most 4 seconds. Their worst enemies are very resistant, well armored invertible robots.

Hammerbots

Hammerbots are robots with hammers or axes that hit their opponents' top. Usually with 4 wheels, their attack is similar to the one from overhead thwackbots, however the weapon actuation is independent of the drive system. The weapon can be fired repeatedly and quickly. It is usually pneumatically powered to deliver enough speed in its course, which has only 180 degrees. The weapon system can work as a mechanism to flip back the robot itself. They are very efficient against robots with weak top armors. Powerful hammerbots are good against rammers, wedges, thwackbots and sawbots. Their worst enemies are the spinners.

Clampers

Clampers are robots capable of holding and lifting an opponent, usually carrying them to the dead zone on the borders of the arena. They're usually pneumatically actuated (faster), or they use an electric system with high gear reduction (slower). Their design strategies are similar to the lifters', where the robot weight should be shifted back to avoid tipping forward when lifting the opponent. Clampers need to be fast enough to reach their opponents before they can escape from their claws. They are good against rammers, wedges and thwackbots. Hammerbots should be caught from their sides, so the clamper can avoid being repeatedly hit by the hammer while clamping them. Instead of a lifting platform, a few clampers use a *dustpan*, which is basically a wide box open at the front and top where an opponent is maneuvered into. A few dustpan designs do not include a restraining claw.

Crushers


Crushers are robots with hydraulic claws capable of slowly puncturing or crushing the opponents. The claws need to have long tips to penetrate efficiently, and they need to have a long course to be able to work against an opponent with large dimensions. Their main advantage is that it is almost impossible for the opponent to escape after being caught, ending the match. Crushers need to be hydraulically powered to generate enough forces to crush, which makes them very complex and heavy, leaving little weight left for the drive system. They are usually heavyweights or superheavyweights.

Flamethrowers


A few competitions allow the use of flamethrowers. The *flamethrowers* are usually used together with other weapons, such as wedges. The effect is mostly visual, counting points with the judges and making the audience cheer. They are usually inefficient to disable other robots because most opponents are fireproof, except if the electronics is exposed or the wheels are flammable.

Multibots

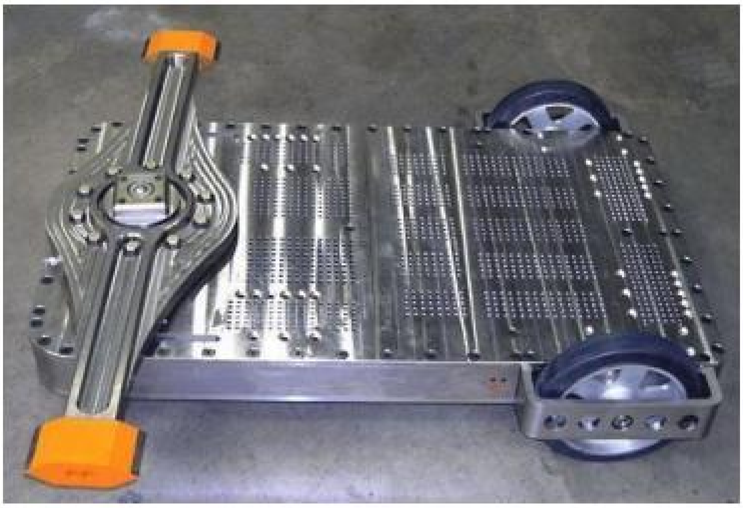

Multibots are robots made out of 2 or more sub-robots, with weights that must not add up beyond the limit of the category. Most of the competitions adopt the rule that says that it is necessary to incapacitate 50% or more (in weight) of the robot to win a round. Using 2 sub-robots is therefore risky, because it is enough to have the heavier one incapacitated to lose a match. For that reason, several multibots use 3 robots of similar weights, forcing the opponent to incapacitate 2 of them to win. For instance, you can use 3 middleweights, as long as one of them drops to 100lb, to compete as a single super-heavyweight multibot (120 + 120 + 100 = 340lb). In the same way that several small weapons are less efficient than a large one, multibots have little advantage over their opponents, unless the attack (usually controlled by 2 or more drivers) is very well coordinated. In practice, it is difficult to coordinate a simultaneous attack, the opponent ends up incapacitating the multibot one by one (in general going for the smallest one in the beginning of the match). Another technique is to use, for instance, a main robot with about 90% of the weight of the category and 2 small ones with 5% each, which serve as a distraction for the opponent. In practice, the small ones are ignored and the opponent goes for the bigger one (the multibot Chiabot used 1 small robot as a distraction, but it didn't help much in practice). Another idea is to use a swarm of small autonomous robots, which would climb the opponent, get inside and destroy them from the inside out. But they are still science fiction, like the Sentries from Matrix, or Star Wars' Buzz Droids.

Design Steps

After choosing the weight class and type of the robot, the next concern is with its cost.

Cost

A middleweight robot, to be competitive internationally, has a typical cost of about US$4,000, including the radio control and spare batteries. For lightweights, about US$3,000 [10], for heavyweights US$6,000, and for super-heavyweights US$8,000. The numbers can go much higher than that. The robot Buster (on the right is a beautifully designed superheavyweight, all made in milled titanium, with an estimated cost of about US$30,000. This doesn't mean that it is not possible to win an international competition with a much less expensive robot, everything depends on creativity. But, statistically, the above numbers are reasonable estimates. The opposite is also true, there is no guarantee that an expensive robot will win a competition.



In summary, this is not a cheap sport. However, for many sponsors such numbers are low if compared with what it is usually invested in other sports. In addition, featherweights and other lighter robots can be quite inexpensive.

Sponsorship

A few great tips about sponsorship and several other combot subjects can be found at The Robot Marketplace (http://robotmarketplace.com/tips.html). Basically, it says that it is not an easy task to find a sponsor that will help you out if you haven't built any combot before. Probably the only exceptions are companies whose owners or directors you know well or who are your friends. Most big companies do not bother with sponsoring robots, it's a better bet to look at smaller local shops near you that might like to help out. You might be able to get sponsorship from big companies, but it is important to meet the right people, the ones who are able to make the decisions. For instance, presenting your robot to a public relations intern won't help you a lot, he/she won't probably be as enthusiastic as you would be when presenting the proposal to their boss. Also, you have to call and visit them in person, nobody will give

you sponsorship over e-mail. Bring with you business cards with your team's logo, for a more professional look, as pictured to the right.

Prepare a presentation folder with lots of nice photos, such as the one pictured to the right Clearly show the potential sponsors how and where their name and logo would be made visible, such as in a Tshirt layout (pictured below), on the robots, at the team website, in YouTube videos, etc. Show as well which newspapers, magazines and TV news programs have already covered the events you plan to attend. Showing videos from the fights is also a great idea, several potential sponsors have no idea of what robot combat is. They might fall in love as soon as they watch it.

Attached to the presentation folder, you should include your annual budget. Don't forget to include the cost of parts, machining time, taxes (especially if any component must be imported), marketing material (such as T-shirts with sponsor logos), event entry fees, and travel expenses. Do not cut down expenses at this stage, ask for everything you might possibly need – there's a chance you get full sponsorship for that value. If you ask for too little in the beginning, you might not be able to increase the budget later on during the same year.

But let the potential sponsors know that they don't need to provide the entire budget, that you will take partial sponsorship. You may even come up with sponsorship levels, such as bronze sponsorship for 10% of the budget, silver for 25%, gold for 50%, and platinum for 100%.

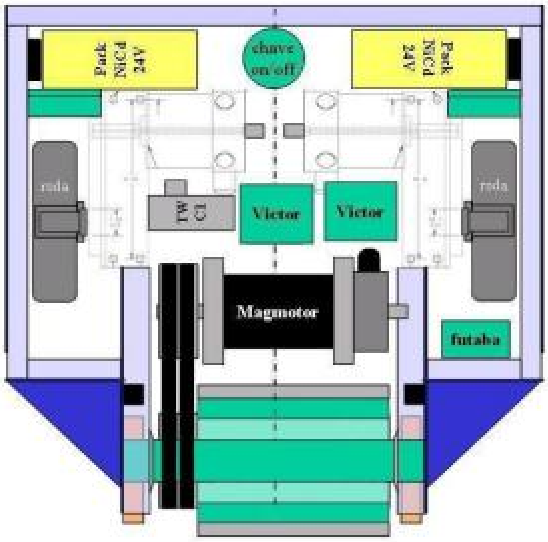
It is important to show the potential sponsors which benefits they get depending on the sponsorship level. For instance, in the T-shirt layout from the previous page, a gold sponsor would get advertising space on the areas 1, 2, 3, 6 and 9, while silver would get 4 and 7, and bronze would get 5 and 8. Needless to say, a platinum sponsor would get all areas from 1 to 9. Note that area 3 is better than areas 4 or 5, because it has a higher chance of being caught on camera during a TV interview, as pictured to the right. Usually, the silver sponsor logo in area 4 is only partially shown during an interview.

Any sponsorship help is welcome. Unless you are very well established with your sponsors, you will find it difficult to get cash from them, more often they might contribute with parts or machining time. And don't give up after getting turned down a few times, you need to put a lot of effort into it.

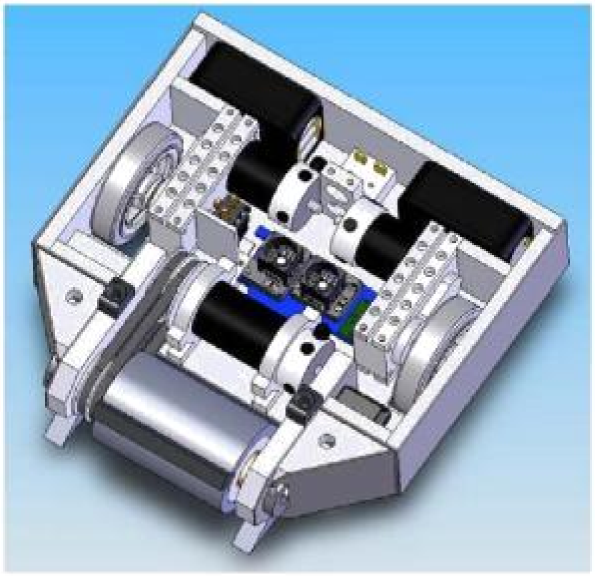
Designing the Robot

The next step is to get an estimate of the robot weight. If, after adding all the motors, wheels, structural components, weapons and batteries, the robot is way above its weight class limit, this means that it is necessary to reduce the entire scale of the robot or to use lighter components. To distribute well the robot's weight, a very useful tip is to use the 30-30-25-15 rule [10]: 30% of the robot weight should be devoted to the drive system (motors, transmissions and wheels), 30% to the weapons (weapon, motor, transmission), 25% to the structure and armor, and 15% to the batteries and electronics. Of course those numbers can vary a lot depending on the type of the robot, but they are representative average values.

When designing and sketching the robot, always have in mind the principle known as KISS: Keep It Simple, Stupid! In other words, don't complicate your design too much if not necessary, design your robot in the simplest possible way, but never simpler than that. Sketches can be made by hand, using a CAD program, or in any way that makes it quick to update and share it with all your teammates. The first sketch of our middleweight Touro was made, believe it or not, in MS Powerpoint, see the figure to the right. It is a program that the entire team had in their personal computers, either at home or in the University, unlike most CAD programs. In this way, the entire team could think anytime anywhere about improvements in the robot design, using any personal computer. This technique is also known as PAD (Powerpoint Aided Design), making it easy to generate vaporbots, which are virtual robot designs that haven't been built yet. Vaporbots help a lot to stimulate creativity and to evolve your design without any building cost.



Here are 4 vaporbots that helped generate the RoboGames 2006 version of Touro.



If you have access to CAD programs such as Solidworks or Rhino3D, then you can use them to create a 3D view of the robot, see the figure to the right (created using Solidworks). CAD programs are also useful to make cutting and drilling marks: just print out the layout in 1:1 scale, glue it directly onto the piece/plate with an adhesive spray (such as Spray 77), and mark the holes with a center punch.

During the design phase, it is necessary to have in mind that fragile items such as electronic components should be placed well inside the robot, to be protected from cutting weapons. The robot should also be the most compact possible, so that its armor can have larger thicknesses without going over the weight limit. But don't forget that too compact robots are difficult to repair during a pitstop, the parts that need to be changed may be inaccessible, so it is important to use common sense.

Calculations

After the first sketches, it is recommended to perform a stress analysis to calculate the resistance of each component from the robot. This subject is too vast, it is beyond the scope of this tutorial. Books about mechanics of solids and mechanical behavior of materials [8] are very useful for that. The analysis consists basically of calculating the tensile, bending, torsion and shear stresses in the structure and components, including the stress concentration factors of the eventual notches (such as holes, abrupt changes of geometry), and combining them to obtain an equivalent stress, usually known as Mises or Tresca stresses. With the equivalent stress, it is possible to design the parts against yield, rupture, plastic collapse, fatigue, etc. Finite element software, such as Abaqus, Ansys, Nastran, Adina, or several others, can be used to aid in the numerical calculation of the robot's resistance. Most of them are capable to import drawings directly from CAD programs. Their license is usually expensive, however these programs are not indispensable. With a little common sense and mechanical background it is possible to make "back of the envelope" stress analyses, which are approximate but accurate enough for design purposes. Chapter 6 will show a few examples of such dimensioning techniques.

Optimization

Most combat robots are born overweight. You must prepare yourself to deal with that, sooner or later. If it is too much overweight, you might need to redesign it completely. Otherwise, a few optimization techniques can be used to lose weight, improve strength, or even to do both at the same time.

One way to do that is to optimize the shape of the robot parts. This is usually done in an ad-hoc manner, using common sense, and sometimes even with the aid of finite element software to check the resulting strength, such as in the spinning disk of the middleweight Vingador (pictured to the right). The holes and voids in the disk were positioned not too close to its center, where outofplane bending stresses can get very high, and not too close to the outer perimeter, to avoid

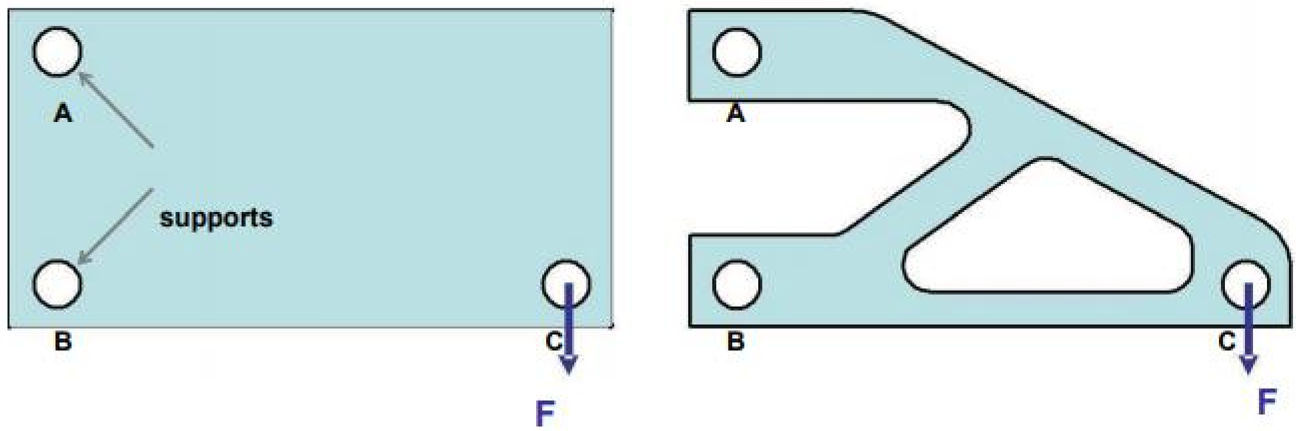
lowering the moment of inertia or the strength of the teeth. This process usually involves trying several hole configurations, and using finite element and CAD programs to calculate the disk strength and moment of inertia.

Shape optimization can also be seen in the spinning bar of the hobbyweight Fiasco, pictured to the right. Pockets were milled in the bar to relieve weight, except at the middle section, not to compromise strength, and at the ends, not to compromise its moment of inertia.

The lifter arm from BioHazard's four-bar mechanism (pictured to the right) is another example of clever shape optimization to selectively remove weight. Note that the weight saving holes near the middle pivot, where the bending moments are maximum, have smaller diameters not to compromise strength. The diameters of the holes are also directly proportional to their distance to the middle pivot, trying to evenly distribute the stresses at the bar, because the bending moment in this system is directly proportional to the distance to the ba r ends.

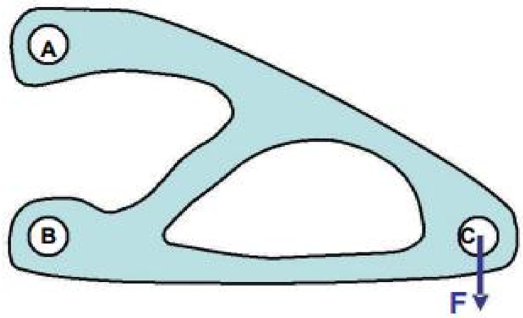
Such optimization tasks can also be performed automatically. Sophisticated software can perform shape and topology optimization of structural parts, to minimize their weight or maximize some property. Shape optimization software like Tosca (http://www.fe-design.de) can be run together with finite element programs to find the optimal shape of a part that will minimize its weight while achieving desired values for stiffness, strength or even moment of inertia, for instance.

For instance, suppose you need to design a single-piece bracket to be fixed inside the robot by two holes A and B, to support some vertical force F that acts at another hole C, as pictured in the next page to the left. The optimization program will require you to inform the relative positions of the holes, their diameters, their contour conditions (such as whether they allow rotations, as if attached by pins, or whether they don't, as if attached by keyed shafts), the bracket material, the direction and intensity of all the applied forces and moments, and the performance requirements. These requirements can be, for instance, the maximum allowable stress in the bracket (a strength requirement) and, at the same time, its maximum allowable deflection (a stiffness requirement), while minimizing its weight. The optimization programs usually require you to inform as well the topology of the component, which is basically the number of voids it may have. And a few programs also allow you to minimize weight together with manufacturing complexity as well, trying to achieve optimum shape using only straight lines and circular arcs, avoiding generic curves.



The figure above to the right shows the resulting shape for minimum weight with minimum manufacturing complexity for a version of our bracket with only one void (besides the voids from the fixed holes A, B and C). Note that this resulting shape is only optimal for specific input values, because it depends on all the given parameters. If,

for instance, the maximum allowable deflection increases while the maximum allowable stress decreases, the shape will be different. Also, if you turn off the minimum manufacturing complexity requirement, you might end up with an even lighter bracket (such as the one pictured to the right), but you'll probably need a numeric control laser or waterjet cutting system to fabricate the resulting intricate part.



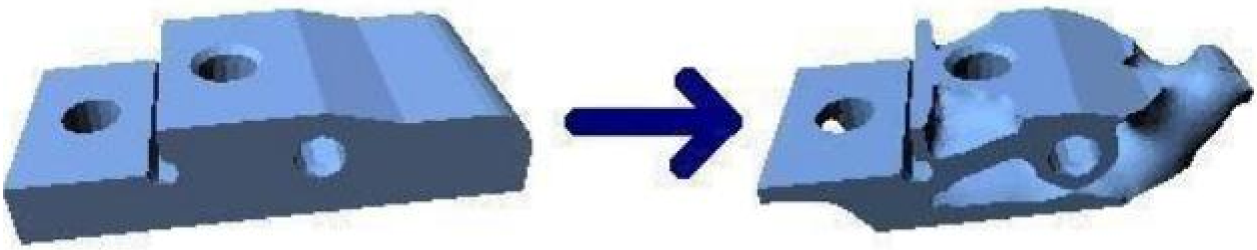
A few programs are also able to optimize both shape and topology, finding not only the shape but also the ideal number of voids in the component. This can be useful, for instance, to find optimal number of voids and their shapes for a spinning disk with maximum strength-to-weight and moment of inertia-to-weight ratios.

Pictured below are a few examples of bracket topologies with 1, 5 and 7 voids (not counting the voids from the holes A, B and C). The above results were obtained after choosing the 1-void topology seen below. A topology optimization program wouldn't need such user choice, it would find out by itself which topology would be the best option, and then optimize its shape.

Note that the topology representations above look a lot like trussed structures, but they'll result in single-piece components, such as in the plates the form the structural frames of the hobbyweight Fiasco (pictured below to the left) and lightweight K2 (to the right). Note also that, for armor plates or other external unprotected structural elements, you'll probably want to turn off topology optimization to force a solution with zero voids. Armor plates with voids would probably be a bad idea against spearbots and flamethrowers.

The topology and shape optimization analyses are not limited to planar problems such as parts with uniform thickness. They can also obtain the shape of optimized tri-dimensional (3D) parts, as pictured to the right. Laser or waterjet cutting won't be enough to fabricate these optimal 3D parts, you'll probably need a mill or even a CNC system.

The 3D optimization process is quite similar to the planar case. You'll feed the software with an initial guess of the shape of the desired component (as pictured below to the left), along with all the required holes and contour conditions, material information, applied forces and moments, and performance requirements. The software will then optimize the topology of the component, adding voids if necessary, and finally output the optimized shape that meets the requirements with minimum weight (as pictured below to the right).



Another approach to optimize the robot is to change the material of its components. Material optimization, either to improve performance or to reduce weight, is seen in detail in chapter 3.

Building and Testing

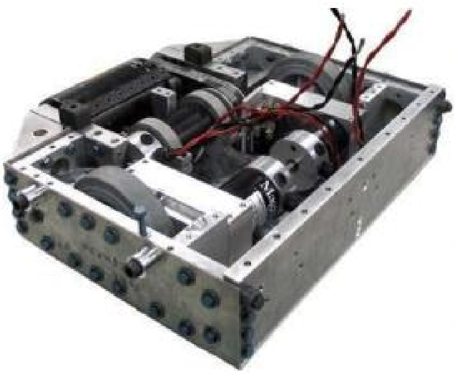
During the robot design, building a full scale model is also very useful. We've already built several models of our robots and their components. For instance, while we were waiting for an Etek motor to arrive in Brazil during its import process, we've built a one-to-one scale model (pictured to the right) using Styrofoam, cardboard, and an old snorkel. These models guarantee that your hand will fit everywhere inside the robot, which is fundamental for quick pitstops. Unfortunately, Solidworks doesn't allow you (yet) to reach your hand inside the monitor. Always recalculate the robot weight, combots tend to easily get overweight.

At least in our experience, we've realized that the design phase usually takes most of the robot development time, perhaps about 60% or more. The other 40% would be the construction itself. In order not to waste money and material, it is a good idea to make sure that the design won't suffer huge changes before starting to build it (small changes during construction will almost always happen). Check the CAD drawings - or the cardboard prototypes - before starting to cut metal. Follow the "measure tw ice, cut once" rule. A lot of information on building the robot will be covered in the following chapters of this tutorial.

Finally, after finishing the robot, there's the part that everybody forgets about (including us): testing. Follow Carlo Bertocchini's law: "Finish your robot before you come to the competition." Many times the robot is finished just before the competition, leaving not enough time to test it. This is a fatal mistake, there are several things that can go wrong. With a few tests most problems can be identified and corrected. Besides, during the tests the driver is able to acquire experience in driving that specific robot, which can make all the difference during a match. This leads to one of Judge Dave Calkins' main advices: LTFD – Learn To *Freaking* Drive! Drive a lot. Hundreds of hours, not a few. Several opponents drive maybe two hours a day. This can and will make a huge difference.

Robot Structure

As for the robot structure, the three main types are: the trussed, the integrated and the unibody. The trussed robots (such as The Mortician, pictured to the right) are built using several



bars, in general welded together, resulting in a very rigid and light structure. The armor is made out of several plates, usually screwed to the bars, sometimes using rubber sandwich mounts (see chapter 4) to provide damping against impact weapons. They are the fastest type of structure to build, it is enough to use a hacksaw and welding equipment to quickly assemble the chassis. Trussed robots are also easy to work with during pitstops because, if one of the plates gets damaged, it is easy to unscrew it and change it for a new one. The greatest disadvantage is to depend on the welds, which are in general the weak point. Besides, the armor plates are prone to be ripped off in combat.

The integrated robots (such as our middleweight Touro, pictured to the right) receive such name because the structure and armor are integrated into a single set, using screws or welds. The same plates that work as armor are the ones where the internal components are mounted to. Sometimes there is a thinner armor layer on top of the integrated structure. Building such robots is not an easy task, however they generate very compact and resistant systems.

The unibody robots (such as our beetleweight MiniTouro, pictured to the right) have their structure milled out of a single solid block. Through milling, it is possible to create the side walls, the bottom, and pockets to fit batteries, motors, etc. In this way, it is not necessary to weld or to use screws in the structure, except to install the components and to attach the top cover. These are the lightest and most resistant robots. However, you lose about 80% to 90% of the material from the solid block to carve its interior, not to mention the hours (or days) hogging the milling machine. The cost and material w aste makes this solution attractive only to very light robots such as the insects. Another disadvantage is that there is no way to replace a damaged part of the structure, as it is done with the armor plates from the trussed robots. If there's too much damage, it might be necessary to mill an entirely new unibody.

A unibody can also be made out of a composite frame, as in the hobbyweight VD2.0 (pictured to the right). Composite frames are basically a foam body with the shape of the unibody, covered with some fiber such as glass, carbon and/or aramid (Kevlar) fibers, which are epoxied to the surface. Carbon fibers are an excellent choice to obtain very high stiffness, while Kevlar gives high impact toughness, see chapter 3. Composite frames are not very popular because they're expensive and difficult to manufacture, in special if the robot design requires the structure to have a high precision to mount, for instance, weapon systems.

Robot Armor

There are basically three types of armor: traditional, ablative and reactive, presented next.

Traditional Armor

Traditional armor plates are usually made out of very tough and hard materials that try to absorb and transmit the impact energy without getting damaged. The high hardness of the armor plate is used to break up or flatten sharp edges from the opponent weapon (which is good against very sharp horizontal spinners), while its high toughness allows the plate to withstand the blow without breaking. This sometimes can be achieved using a composite armor, which means using several layers of different materials. For instance, you can use very hard (but brittle) ceramic tiles sandwiched between two very tough (but relatively soft) stainless steel plates to use as armor.

Due to their high hardness, traditional armors need to be changed less often, and they look nicer after a match. However, traditional armors transmit a lot of the impact energy to the rest of the robot structure, as shown in chapter 6, and they usually produce sparks, which may count as trivial or cosmetic damage points depending on the judges.

Ablative Armor

Ablative armor plates, on the other hand, are designed to negate damage by themselves being damaged or destroyed through the process of ablation, which is the removal of material from the surface of an object by vaporization or chipping. They're also made out of tough materials, but with low hardness and low melting point to facilitate the ablation process.

Ablative armor plates are much more efficient dissipating the impact energy, which is mostly absorbed by the ablation process, transmitting much less energy to the rest of the robot. They are a good choice especially against blunt or not-so-sharp horizontal spinners. They are also good against drumbots, even the ones with sharp teeth, because most of the drum energy will be spent "eating out" chunks of the armor instead of launching the robot. Also, you won't get sparks if you're using an aluminum ablative armor, which is good even though they are only counted as trivial damage.

Thick wooden plates are also very efficient as ablative armor, however they result in a lot of visual damage that may award damage points to your opponent (even though the destruction of ablative armors should only count as cosmetic damage). Make sure that the judges know if you have an ablative armor. A disadvantage of ablative armors is that they need to be changed very often because of the ablation, and you might need to use gloves to handle your deeply scarred robot.

Reactive Armor

The third armor type is the reactive. Such armor reacts in some way to the impact of a weapon to prevent damage. Most of them are very effective against projectiles, which have a relatively low mass and extremely high speeds, but not much against regular combot weapons, which have much more mass but much lower speeds than projectiles.

One example is the explosive reactive armor, which is made out of sheets of high explosive sandwiched between two metal plates. During the impact, the explosive locally detonates, causing a bulge of the metal plates that locally increases the effective thickness of the armor.

Another example is non-explosive reactive armor, which basically consists of an inert liner, such as rubber, sandwiched between two metal plates. Against most combot weapons, this basically works as a shock-mounted armor, dissipating energy in the elastic liner. And against spearbots this armor has an additional advantage: during an angled impact, the outer metal plate will move laterally with respect to the inner plate, which may deflect or even break up any spear that eventually penetrates.

There are also studies on electric reactive armors, which would be made up of two or more conductive plates separated by air or some insulating material, creating a high-power capacitor. This could be implemented in practice using three metal layers, separated by rubber liners (acrylic tape such as VHB 4910 is also a good option due to its high dielectric breakdown strength), which would also work as a shock-mount. The middle plate is then charged by a high-voltage power source, while the other 2 plates are grounded. When the opponent's weapon penetrates the plates, it closes the circuit to discharge the capacitor, vaporizing the weapon tip or edge, or even turning it into plasma, significantly diffusing the attack. Note, however, that this system might be very difficult to implement in a combat robot, not to mention the increased battery requirements. Also, most competitions forbid the use of electric or explosive reactive armors.

Robot Drive System

The three usual types of drive systems are based on wheels, tank treads and legs, discussed next. There are also other types, based on rolling tubes (moving as a snake), rolling spheres, or air cushions (hovercrafts), but they're not very effective in combat. Flying is usually forbidden.

Tank Treads and Legs

Robots with tank treads are beautiful, they have excellent traction, however they waste a lot of energy when turning due to ground friction. They are also slow when turning, which allows an opponent to drive around and catch them from behind. Besides, treads can be easily knocked off by opponents with powerful weapons.

Legs have also several disadvantages. They are complex to build and control, and they usually end up not sturdy enough for combat, in special against undercutters. They tend to leave the robot with a high center of gravity, making it easy to get flipped over. Their only advantage is the weight bonus, usually 100%, allowing for instance a 240lb legged combot to compete among 120lb middleweights. But note that shufflers, which are rotational cam operated legs, are not entitled for the weight bonus. Legs are usually good options for robots in very rough and uneven terrain, which is not the case in flat floored combat robot arenas. This is why the international combat robot community has converged to the wheel solution.

Wheel Types

There are several types of wheels in the market. A few robots use pneumatic wheels, however they are filled internally with polyurethane foam so that they don't go flat if punctured.

Another good solution is the use of solid wheels. To maximize traction, it is recommended that solid wheels have an external layer of rubber with hardness around 65 Shore A, at most 75 Shore A. Harder wheels tend to slide. Wheels with hardness measured in Shore D units are probably too hard.

Several robots, as well as our middleweights Touro and Titan, use the Colson Performa wheels (pictured to the right). These wheels are very inexpensive: each 6" wheel from Touro costs only US$7.25. Besides the low price, this Performa model from Colson has hardness 65 Shore A, a very good value for traction.

An interesting wheel solution was used by our team during the RoboCore Winter Challenge 2005 competition, held on an ice arena. When driving on ice, the wheel hardness is irrelevant. The important thing is the presence of sharp metal tips to generate traction. The secret of walking on ice is to know that it is not friction (very low in this case) that generates traction, but normal forces. The solution for the problem was very cheap: we've inserted several selfdrilling flat head screws at angles of about 60 degrees with respect to the wheel radius (as pictured to the right). The screw caps were also sharpened to improve traction performance. Those sharp tips generate a very small contact area with ice, generating a very high contact pressure. That high pressure makes the ice melt locally, allowing the tips to slightly sink in and lock in place. Then, when the motors spin the wheels, the "fixed" tips sunk in the ice apply normal *horizontal* forces, generating traction without sliding. The traction on ice ends up even better than the one from a regular wheel on metal. See in the picture above that we chose to use a single row of screws: our tests with 2 parallel rows generated worse traction, because with twice the number of screws to distribute the load, the pressure on the ice drops in half, and the screws sink in much less. A single well sunk screw generates much better traction than two half-sunk screws.

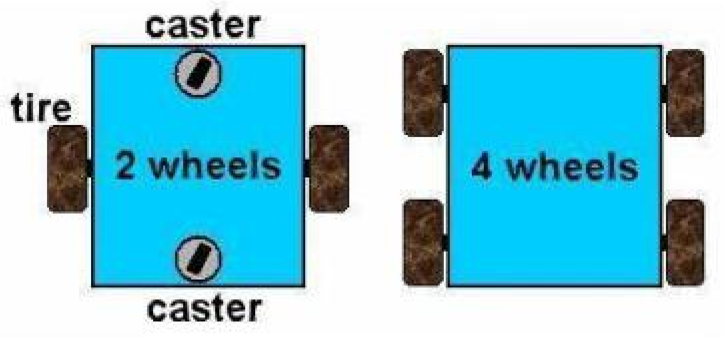
Also notice that we've alternated the screw angles on the wheel, to guarantee that in average the traction was identical in both forward and reverse directions.

Wheel Steering

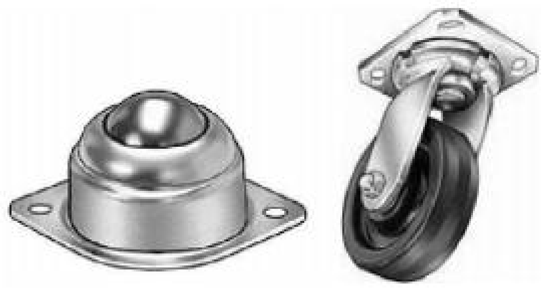
There are two main types of wheeled vehicles: the ones with Ackerman steering and the ones with tank (or differential) steering. Ackerman is the solution adopted by the automobile industry: a large motor is used to move the system forward or backward, and another smaller motor controls the steering of the front wheels to make turns. This is very efficient for high speeds, because it is easy to drive straight, however it demands several maneuvers for the robot to spin around its own axis. Besides, the steering system is usually a weak point, needing to be very robust and, consequently, very heavy.

Tank steering receives that name for being used in war tanks. The entire left side of the robot is driven independently from the right side. To drive on a straight line, it is necessary that both sides have the same speed, which is not always easy to guarantee. Turns are accomplished when those speeds are different. The great advantage of that method is that if the speeds of both sides are equal in absolute value but have opposite senses, the robot can turn on a dime. This is perfect to always keep facing the opponent.

Two -Wheel Drive



There are two common options in tank steering, which are using 2 or 4 active (power driven) wheels, as pictured to the right. With 2 active wheels, it is possible to turn very fast and with less waste of energy. In addition, the robot saves weight by not needing the extra set of active wheels, shafts and bearings.

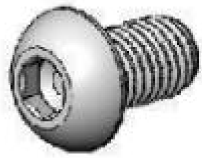


With only 2 active wheels, the robot will probably need at least another ground support, ideally 2. This is usually accomplished with skids, which are passive elements such as ball transfers or caster wheels (pictured to the right). Try to place the axis of the 2 active wheels as close as possible to the robot center of gravity, and the ball transfers or casters in the front and in the back, arranged in a cross configuration (see the previous figure for a 2-wheeled robot). In this way, you guarantee that almost the entire reaction force from the ground will go through the 2 active wheels, where you need traction. Our middleweight Ciclone, due to lack of internal space, could not use the cross configuration. The 2 active wheels ended up in the back, as pictured to the right, supporting only half of the robot weight, compromising traction. But be careful with the cross configuration, make sure that the ball transfers or casters won't lift the active wheels off the ground, e specially in an arena with uneven floor. To avoid lifting the

active wheels and losing traction, these robots should have their passive elements in a plane a couple of millimeters higher than the active wheels. You can also spring mount the ball transfers and casters, creating a suspension system that prevents the active wheels from being lifted off.

A disadvantage of using only 2 wheels is that it is more difficult to drive on a straight line. Several electric DC motors do not have neutral timing, which means that they spin faster in one sense, making it harder to drive on a straight line. If possible, try to set the drive DC motors in neutral timing (see chapter 5) or, if the radio control system is programmable, try to compensate the speed differences through the trim settings. If your robot continues with problems to move on a straight line, try to use rigid casters (pictured to the right) instead of the swivel ones. You will have a harder time turning, but the robot will drive straighter.

In robots with very violent weapons, the ball transfers and casters may not take the extreme forces transmitted to the ground during an impact against the opponent (these forces can easily exceed a few metric tons for middleweights). In this case, you can replace these passive elements by, for instance, button-head cap screws (pictured to the right), mounted upside down at the bottom of the robot. The round head slides very well on the arena floor. This is the technique that we use in our middleweight spinner Titan. Use hardened steel screws, the ones with high class (see chapter 4), because they are harder and do not easily wear away due to friction with the arena floor. A few robots also use wide pieces of Teflon (PTFE) for ground support with reduced friction.



All -Wheel Drive

Another wheel steering option is the use of 4 (or more) active wheels. Four-wheeled robots drive better on a straight line, they are good against wedges and lifters (because in general they guarantee at least 2 wheels on the ground to be able to escape if they've been lifted), and they have redundancy in case a few wheels are destroyed during a match.
Experienced drivers, such as Matt Maxham from Team Plumb Crazy, are still able to drive even after 3 out of 4 wheels have been knocked off!

A few robots use 6 or 8 wheels to maximize traction as well as to increase redundancy, such as the 8-wheeled super-heavyweight New Cruelty and the famous 6-wheeled heavyweight Sewer Snake, pictured to the right. The problem with 4 or more wheels is the waste of energy while making turns, besides the additional drivetrain weight needed by the additional axes, bearings, pulleys, etc.

An interesting solution to help 6-wheeled robots to make turns more easily is to have two middle (compliant) wheels with a slightly larger diameter. In this way, the ground normal forces on the 4 outer wheels are reduced.

Omni -Directional Drive

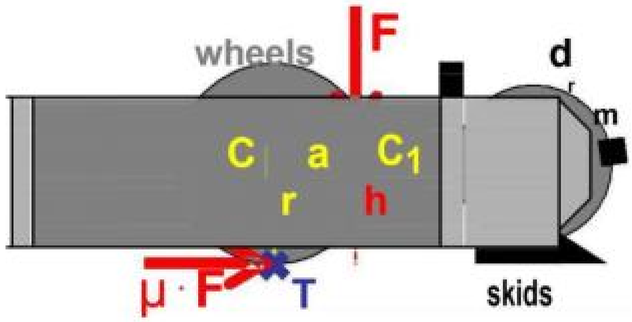
A very specific type of drive system is the omnidirectional drive. It can be accomplished with omnidirectional wheels (a.k.a. Mecanum wheels), which can move sideways without changing their direction. Those wheels have several small passive rollers, which can rotate freely, as pictured to the right.

The most common configurations are 4 parallel wheels, or 3 wheels at 120^{O} angles. The rollers provide the wheel with traction in only the circumferential direction, rotating freely in the shaft axial direction. Coordinating the movement of the 3 (or 4) wheels, it is possible to move sideways without changing the direction of the robot. In the case of 3 wheels at 120^{O}, the omni-directional control system is not so simple to implement, you need to program a few calculations involving sines and cosines. An off-the-shelf solution is the OMX-3 Omni-Directional Mixer (pictured to the right), a small US$45 board from Robot Logic (www.robotlogic.com), which does all these calculations automatically. This system is excellent for robot soccer competitions: the robot with the ball can move sideways to dribble an opponent without losing sight of the goal. It is possible to kick towards the goal immediately after dribbling the opponent, without wasting time changing direction and making turns.

However, in combat robots such omni-directional capability is not necessary, because during a match you do want to be pointed towards your opponent. This is usually your goal. Moving sideways can be a good idea to dodge from an attack, but the cost-benefit is not good: the omni-directional wheels have worse traction than regular ones, they are less efficient (they waste more energy), and the rollers don’t stand violent impacts.

Wheel Placement

Another important factor in the design of the drive system is the location of the center of mass of the robot. If it is shifted to the left of the robot, for instance, the wheels on this side will receive a larger reaction force from the ground. Because of that, they would have better traction than the right side, and the robot wouldn't move straight. Try to distribute the weight equally on both sides. For robots with only 2 active wheels, it would be ideal to have the robot center of mass C_1 very close to the center C of the line that joins the wheel centers, as pictured to the right. In that case, each wheel would receive about half the robot weight, guaranteeing good traction.



Even better would be to place C_1 slightly ahead of C, at a horizontal distance $a = \mu h$, where μ is the coefficient of friction between the tires and the ground, and h is the height of C_1, as pictured to the right.

If the combined torque of both wheels is large enough to guarantee an initial traction force of $\mu \cdot F$, the maximum value without wheel slip, where F is the robot weight, then the robot won't tilt backwards while it is accelerating if $a \geq \mu \cdot h$. It won't tilt because the forward gravity torque $F \cdot a$ with respect to the contact point T between the wheel and the ground becomes greater than or equal to the backward inertial torque $\mu \cdot F \cdot h$ (with respect to T) caused by the forward acceleration of C_1, since $F \cdot a \geq \mu \cdot F \cdot h$.

Our middleweight Touro uses this principle. Not tilting backwards is good to prevent wedges from entering underneath the skids that support Touro's drum. In addition, during the initial drivetrain acceleration, 100% of the robot weight goes to both wheels if $a = \mu \cdot h$, maximizing the initial traction force. The front skids will only feel part of the robot weight after the wheel motors drop their traction force below $\mu \cdot F$, which happens as they speed up. So, with $a = \mu \cdot h$, Touro achieves its maximum possible initial drivetrain acceleration while the front skids are barely touching the ground, until it acquires enough speed to make the front skids get some downward pressure right before they get in contact with the opponents, trying to get under them using the sk ids as if they were wedges. But it is important not to make this distance much higher than $\mu \cdot h$ As discussed before, our middleweight Ciclone, due to lack of space, had its 2 active wheels far in the back of the robot (as pictured to the right), away from the center of gravity, resulting in a large value for the distance a. Each wheel ended up just bearing about one fourth of the robot weight, the other half went to the front ground supports. With this reduced applied load on the active wheels, Ciclone had poor traction, with a lot of wheel slip. Note that magnet wheels and suction fans would be two possible solutions, although unusual, to increase the normal forces at the wheels.

The distance between the robot bottom and the arena floor is also important, this clearance needs to be large enough to avoid being trapped in debris or in uneven seams of the arena floor. If your poor featherweight will fight right after the super-heavyweight hammerbot The Judge, it will probably have to overcome arena conditions such as the one pictured to the right. But the ground clearance cannot be too large, otherwise your robot might be vulnerable to wedges, lifters and

launchers. It is also important to keep low the robot center of gravity to avoid being flipped over. In our experience, a minimum suggested clearance for any class from hobbyweight to super-heavyweight is about 1/4” (or 6mm).

Invertible Design

Most successful combat robots are invertible, which means that they can be driven while upside down. With the high number of wedges, drumbots and vertical spinners that we see today, not to mention launchers and lifters, it is just a matter of time until your robot gets flipped over.

If your robot is not invertible, or if its weapon has limited or no functionality when upside down, then it is a good idea to have some selfrighting mechanism (SRiMech). A SRiMech is an active system that returns an inverted robot to its upright state. This mechanism can be an electric or pneumatic arm, or a passive extension on the upper surface of the robot to roll or flip it upright, such as the large white hoop on top of the featherweight Totally Offensive, pictured to the right. Launching or lifting arms, or even vertical spinning weapons, can also be used as a SRiMechs if properly designed.

The easiest way to implement an invertible design is to use wheels that are taller than the robot chassis. If the front side of your robot has a tall chassis, but not its back side, then another option is to use 2 drive wheels in the back, and skids in the front.

Another solution is to have two sets of active wheels, driven together using chains or belts, one of them for driving the robot when not flipped, and the other to be used when upside down. But note that this solution usually adds a lot of weight to your drive system.

Also, it is important to remember that your robot does not have only 2 sides. If it is box-shaped, it actually has 6 sides. But it is very simple to avoid losing a match because your robot ended up standing on its side. You only need to avoid having perfectly flat and vertical front, back and side walls.

This can be accomplished, for instance, using bolts sticking out of the walls, as pictured to the right, circled in red, in our wedge Puminha.

The bolts should not stand out too much, to avoid being easily knocked off, but enough to make sure that the robot will tilt back, as shown in the bottom picture to the right.

If the robot is invertible, without a preferential side, then you can place the screw in the mid-height of the chassis. Otherwise, you’ll probably want to place it near the top of the chassis, as in both pictures to the right, to increase the chance of tilting back in the upright position.

Robot Weapon System

The wide range of weapon systems makes it difficult to give general suggestions that would apply to all of them. So, the weapon system of each robot type needs to be studied on a case by case basis. This topic is extensively covered in chapter 6, which deals with Weapon Design. You can also find several weapon design tips in sections 2.3 and 2.4. In addition, chapter 3 will show a thorough discussion on material selection for all kinds of weapon systems. Chapter 5 also deals with this subject, showing spin-up calculations for kinetic energy weapons such as spinning bars, disks and drums.

There are also very good books on the subject. For instance, Combat Robot Weapons [6] is entirely dedicated to weapon systems. Build Your Own Combat Robot [3] and Kickin' Bot [10] have chapters explaining the design details of each weapon type and the related strategies. And Building Bots [4] even presents basic physics equations. There are also a lot of weapon-related questions and answers at http://members.toast.net/joerger/AskAaron.html.

Note that there are several weapons that are usually not allowed in combat. This includes, but is not limited to, radio jamming, noise generated by an internal combustion engine (ICE), significant electro magnetic fields, high voltage electric discharges, liquids (glue, oil, water, corrosives, etc.), foams, liquefied gases (if used outside a pneumatic system), halon gas fire extinguisher (to stop an opponent's ICE), unburned flammable gases, flammable solids, explosives, un-tethered projectiles, dry chaff (powders, sand, ball bearings), entanglement weapons (nets, strings, adhesive tape), lasers above 1 milliwatt, and light, smoke or dust based weapons that impair the viewing of robots (such as the use of strobe lights to blind the opponent driver).

Building Tools

The following chapters will present the several materials and components necessary to build a combat robot. But for that it is desirable to have access to a series of tools. Below, there is a comprehensive list with everything that could be useful in the construction of a combat robot. Most of the items can be found, for instance, at McMaster-Carr (www.mcmaster.com).

The book Kickin' Bot [10] has very good sections on how to effectively use most of these tools. There's also a great 43-minute video at http://revision3.com/systm/robots, featuring RoboGames founder Dave Calkins, teaching how to use basic tools to build a combot, as well as presenting a primer on the involved components.

Of course it is not necessary to own the entire list of tools presented below to build a combot. If your robot has some special part that you're not able to build by yourself, either due to lack of experience or to restricted access to a machine shop, you can have it machined straight from its CAD drawing through, for instance, the www.emachineshop.com website.

Mechanical

- safety: safety glasses, goggles, face shield, gloves, ear muffs, first aid kit;
- wrenches: Allen wrench (L and T-handle), combination wrench, open-end wrench, socket wrench, adjustable-end wrench, monkey wrench, torque wrench;
- screwdrivers: flathead and Phillips;
- pliers: needlenose plier, cutting plier, vise-grip, slip-joint plier, retaining ring plier;
- clamping: C-clamp, bar clamp, bench vise, drill press vise;
- measuring: caliper, micrometer, steel ruler, tape measure, machinist's square, angle finder, level;
- marking: metal scriber, center punch, automatic center punch, hole transfer;
- cutting: scissors, utility knife, Swiss army knife;
- drilling: drill bit, unibit, countersink, counterbore, end mill, hole saw, reamer;
- tapping: tap, tap wrench;
- hand tools: hacksaw, file, hammer, jaw puller, keyway broach, collared bushing, telescopic mirror,telescopic magnet;
- weighing: dynamometer, digital scale;
- power tools: power drill (preferably 18V or more), jigsaw, Dremel, angle grinder, orbital sander,disc sander, circular saw;
- large power tools: lathe, bench drill, bench grinder, vertical mill, bandsaw, miter saw, belt sander, guillotine, CNC system, water jet system, plasma cutter;
- welding: oxyacetylene, MIG and TIG;
- cleanup: air compressor, air gun, vacuum cleaner (metal bits can short out the electric system).

Electrical / Electronic

- pliers: flush cutter, needle plier, crimper, wire stripper;
- soldering iron and support with sponge, desoldering tool;
- tweezers, magnifying glass, board support;
- digital multimeter, power supply, oscilloscope, battery charger;
- hot air gun, glue gun.

Fluids

- WD-40 (lubricant, it can be used to cut, drill and tap, and to clean Colson wheels);
- stick wax (to lubricate cutting discs);
- threadlocker (Loctite 242, it locks the screw in place);
- retaining compound (Loctite 601, it holds bearings);
- professional epoxy (the 24 hour version), J.B. Weld (even stronger metallic bonds);
- alcohol and acetone (metal cleanup before applying epoxy);
- layout fluid (to paint the parts and later mark holes or draw lines for cutting);
- penetrant dye (to inspect the presence of cracks);
- adhesive spray (3M Spray 77, to glue layout printouts onto plates);
- citrus-based solvent (Goo-Gone);
- solder paste and liquid electrical tape;
- wheel traction compound (Trinity Death Grip).