ROBOTIQ

FORCE SENSORS IN ROBOTICS RESEARCH

THE ESSENTIAL GUIDE

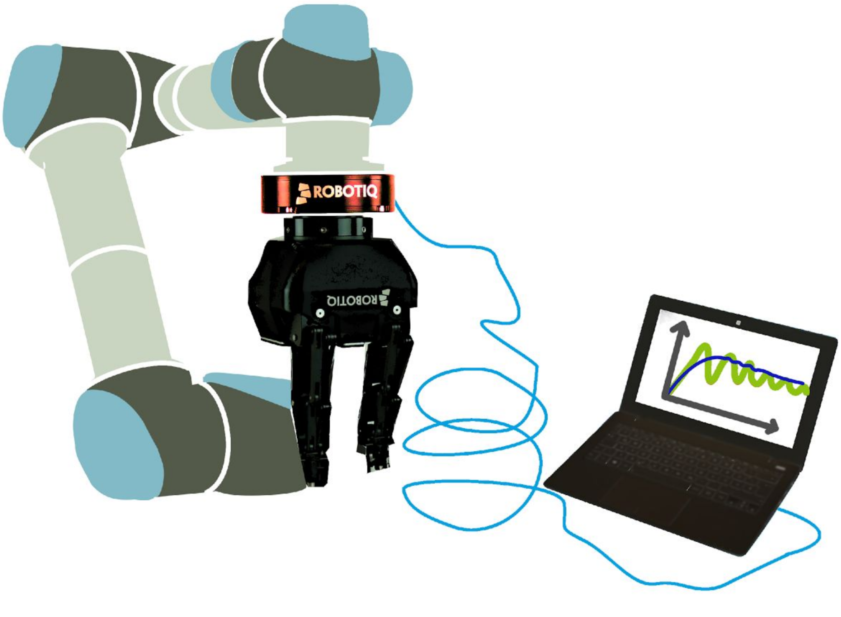


TABLE OF CONTENTS

Published: January 25nd 2016
Author: Alex Owen-Hill
Revision: 0.4

Lean Robotics: Simplify Robot Cell Deployments

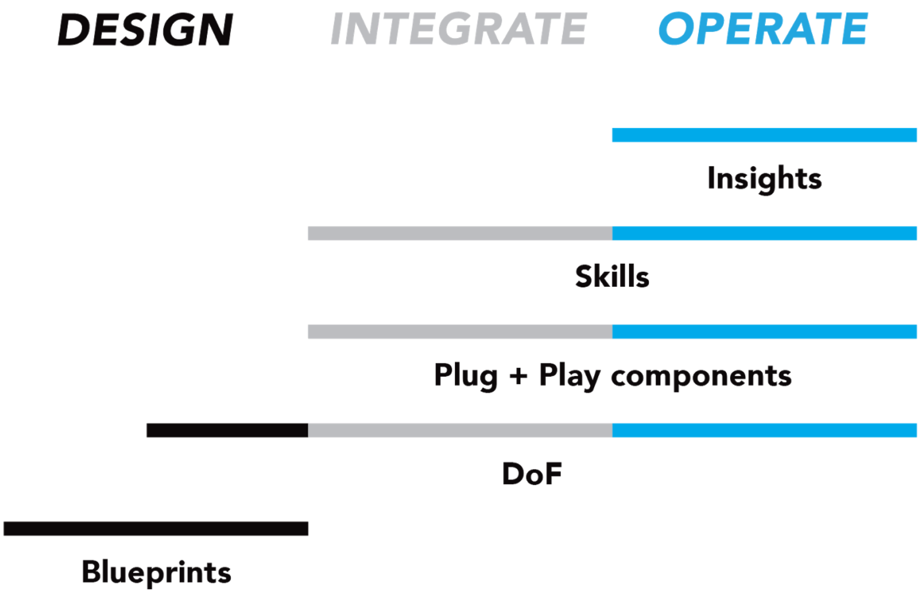
Whenever you ask if robots could work in your factory, the answer you receive is always a hesitant "It depends." It depends on your factory, your team, which robot you choose, what you want it to do... and a whole lot more.

If you're a first-time robot user, how can you get started? How do you get from your initial idea to a productive, working robot? And if you've already got a few robotic deployments under your belt, how can you scale up your robotics efforts throughout your factory—or across multiple factories?

The answers can be found in **lean robotics: a methodology for simplifying robotic cell deployments**.

Lean robotics is a systematic way to complete the robotic cell deployment cycle, from design to integration and operation. It will empower your team to deploy robots quicker and more efficiently than ever before.

Lean robotics divides robotic cell deployments into three phases: Design, Integrate and Operate.



Robotiq's library of eBooks covers the different phases of the robot cell deployment to ensure that you have access to tips from robotics experts all along.

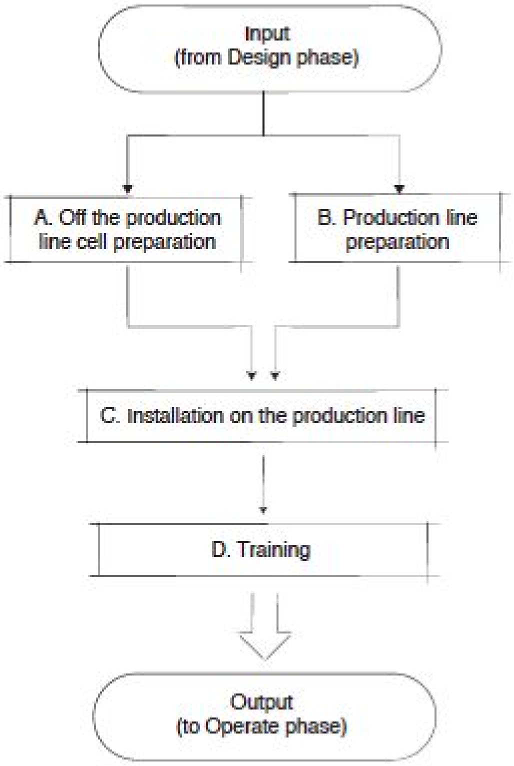
Learn more about Lean Robotics on **leanrobotics.org**.

This Ebook Covers the Integrate Phase

INTEGRATE

The integrate phase consists of putting the pieces of the robotic cell together, programming it, and installing the cell on the production line.

You start the integrate phase with the cell design in hand and the equipment ready to be assembled. At the end of the integrate phase, you'll have a working robotic cell on your production line, ready to start creating value for its customer.



PART 1: CHOOSING FORCE SENSORS FOR RESEARCH

INTRODUCTION

Force sensors are a pretty important part of robotics research.

They provide robots with a way to physically interact with the environment. They ensure that robots don't damage the environment, or harm people, by applying too much force. They provide vital data to verify our research developments through experiments and product tests.

Force sensing has been used in robotics since right at the beginning, with research into force control starting at the end of the 1970s.

You'd think there must be loads of information for roboticists on using force sensors in their research, right?

Surprisingly not.

Perhaps it's because force sensors are so common in our field that there is not much straightforward, clear advice on how to use them in robotics research. Experienced researchers are so accustomed to using force sensors that they forget what it was like to use them for the first time. New researchers end up making the same mistakes over and over again because nobody has given them a guiding hand.

This ebook attempts to solve this problem by giving you information to start using force sensors in robotics research.

It's not an exhaustive guide. That would take up hundreds of pages and probably never be read by anyone. Instead, you can think of this ebook as a starting point, from which you can begin your own further research.

This ebook will allow you to quickly start using force sensors in your research, with some tips on how to avoid common mistakes.

In Part 1, we'll introduce the different types of force sensing technology and how to choose the right one for your research application.

In Part 2, we'll talk about how you can integrate force sensors into your setup and how to get started with force control (if you need to).

In Part 3, we'll look at how you can use force sensors as a way of collecting data for your experiments and how to use them with ROS.

Finally, in Part 4, we'll tell you where you can get even more free information about using force sensors in research, through the email series which accompanies this ebook.

We hope you enjoy it!

WHAT FORCE SENSORS MEAN FOR ROBOTICS RESEARCH

To start with, let's cover the basics of force sensing. What are force sensors? How do they work? What can robotics researchers use them for?

This section will give you a crash-course in force sensing technologies and applications.

WHAT IS A FORCE SENSOR?

As the name suggests, force sensors detect the forces applied between their base and a sensing plate. Force-Torque Sensors (FT Sensors) detect both the forces and the torques. We often attach them to the end of a robotic arm, before the end-effector, to detect interaction forces with the environment. This data can then be used by the robot's control system to improve the operation – for example, by reducing the forces applied to the environment or task. However, in research we can also use FT sensors for some other applications, such as gathering external force data in experiments.

There are a huge variety of different force sensors, ranging from very cheap analogue pressure sensors right up to 6-Axis FT Sensors. We'll briefly cover some of the different types in the next section, but mostly we'll be talking about 6-Axis FT Sensors, which are probably the most common.

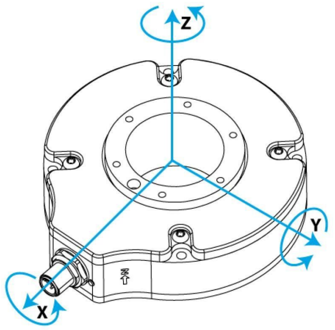
6-AXIS FORCE-TORQUE SENSORS

A 6-axis FT Sensor detects forces in all three Cartesian axes (x, y and z) as well as all three rotational axes (roll, pitch, yaw or α, β, γ). Using this data you can detect any force which is applied to the sensor.

You will need a 6-axis FT sensor if you want to detect how much force your robot is applying to the environment in multiple axes. They are often used in manipulation research to ensure the robot applies a minimal force to the task. They are not tactile sensors, so do not detect slipping forces, and are often physically quite big. Therefore, they are not usually used to detect grasping forces, e.g. on the fingertip. However, they are highly versatile and very common in robotics research labs due to their ease-of-use and good quality data.

HOW DO FT SENSORS WORK?

Many FT sensors, even simple ones, consist of strain gauges which are attached to a piece of material, e.g. steel. When a force is applied, the material bends and the resistance of the strain gauge changes. In the most rudimentary force sensors, you have to measure the changing resistance yourself. However, with most modern FT sensors, the internal electronics take care of all the calculation and signal cleaning. This leaves you with a nice clean digital force signal for all six axes.



6-axis FT Sensors have two ends: a fixed casing and a floating plate. The fixed casing is usually attached to the end of the robot arm, while the floating plate is attached to the end effector or tool. Inside the FT sensor, the plate is attached to the casing by three metal beams. Each beam is fitted with a set of highly precise strain gauges, which measure the deflection of the beam when a force is applied.

Often, FT sensors can be used to hold payloads far beyond the maximum rated lifting force of the robot. For example, the Robotiq FT 300 sensor is mechanically rated at 5x its measurement range of ±300 N in x,y,z (amounting to approximately 1.5kN). When you consider that the UR5 robot (which has out-of-the-box compatibility with the FT-300) has a payload capacity of 5kg (49N under gravity), this is quite a big factor of safety. However, it is important to remember this "mechanical overload" value when choosing a force sensor for your application, especially if you are using the FT Sensor to measure forces applied to a heavy external structure, using a high-payload robot or moving heavy objects under speed (see the section on FT Sensor Specifications below for more details on mechanical overload).

WHAT COULD I USE A FT SENSOR FOR?

The possibilities for using a FT Sensor in your research are only limited by your imagination and practicality. The Robotiq FT 150 has been used in research applications ranging from human-robot collaboration to assembly and force feedback. You can also use FT Sensors for telemanipulation or even exoskeletons.

From a practical perspective, we can group potential applications into two types:

- **ATTACHED TO THE ROBOT —** The FT Sensor is attached the end of the robot and used to measure the robot's interaction forces with the environment.
- **ATTACHED TO SOME EXTERNAL STRUCTURE** — The FT sensor is attached to a fixed location and used to measure forces applied by any external sources: robot, human or anything else.

When you are choosing an application, consider which of these approaches is most suitable and how much development time you have available.

Generally, attaching the sensor to the robot requires more development, unless you are using "out-of-the-box" routines for a known robot. For example, the Robotiq FT Sensors include functionality out of the box for Universal Robots models. Such routines will merge the robot's kinematics with the force measurements to return the x,y,z forces on the environment. If you don't have pre-programmed routines for your robot and force-sensor, you will have to code them yourself.

Attaching the sensor to the environment will give you an easier signal to work with (e.g for use in experiments). However, it is far more limited if you want to detect forces applied in several parts of the workspace.

The best way to come up with a good application is to become familiar with what FT Sensors are (and are not) able to do. Just watch our video to see what they're capable of.

DO I REALLY NEED A FT SENSOR?

You might see other research groups using force control and think that it will be the answer to all of your manipulation problems. This may be true, but you should become familiar with the benefits and limits of the technology. Would position-based control be sufficient for your application?

In research, unnecessarily adding force control to a robot can take a lot of time, especially if you have to develop the force control yourself. Therefore, consider carefully exactly how you want to integrate a FT Sensor into your robotic platform before rushing to implement something you might not need.

If you do decide that you do need to use force control, see Part 2 for tips on how you can get started.

TYPES OF FORCE SENSORS

It's probably not surprising, but there are many different types of force sensor. In fact, there are new technologies being developed every single day. Researchers frequently find new applications which require a different approach to force sensing (such as MEMS Force sensors, FT sensors which work in fMRI machines, etc).

There are so many options that one of the most difficult choices for robotics researchers is:

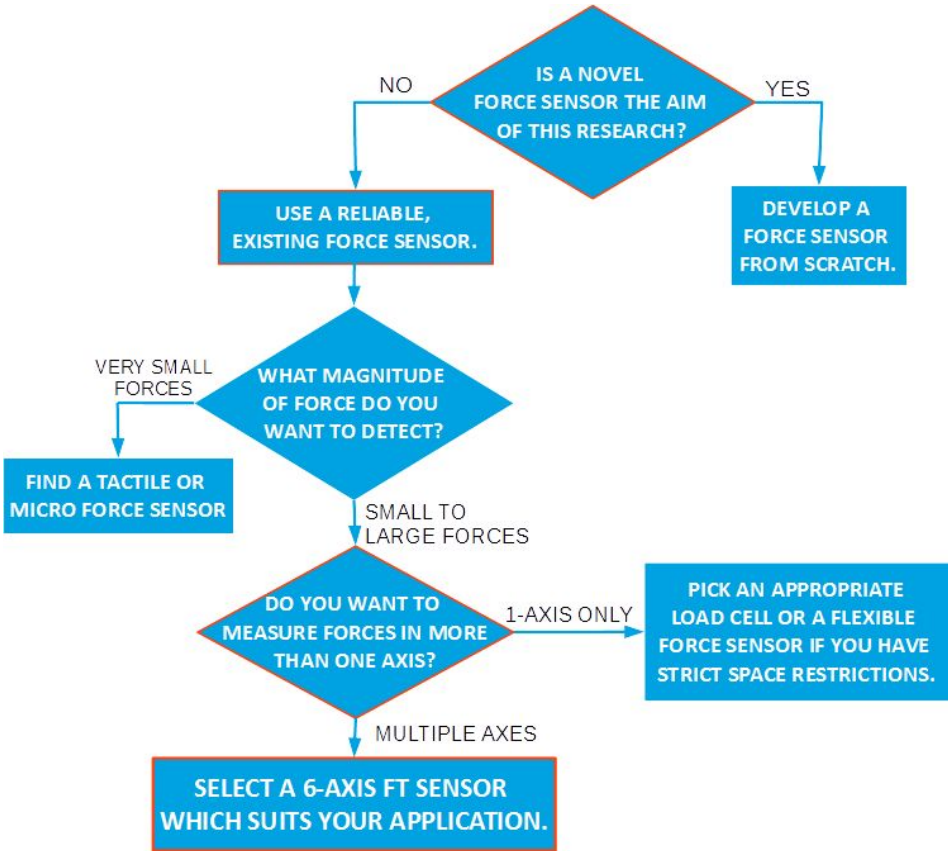
Which sensing technology are we going to use?

In this section, we introduce some common force sensing technologies.

WHAT SORT OF FORCE SENSOR DO I NEED?

As you have picked up this guide (and read this far) I'm going to guess that you already know you need to get a force sensor for your research. Just like any investment, choosing the right force sensor is a balance between cost, time available for integration/development and specifications. We will cover the most important specifications of 6-axis FT Sensors in the next section.

There are a huge variety of force sensors, using a range of different sensing technologies. Given the many different options, it might seem difficult to choose the correct one for a research application. However, in reality the choice is often quite simple, as this flowchart shows



6-axis FT Sensors are so common in robotics research labs for a reason: they are reliable, accurate and very easy to use. The only real reason you should choose a different type is if a 6-axis FT Sensor doesn't suit your application.

CONSIDER THE LONG-TERM IMPLICATIONS

One of the reasons that 6-axis FT Sensors are so popular is that they're quite "future proof." It is important to consider how your force sensor might be used in future research projects. Any force sensor is an investment and you want to makes sure that it does not become just "another piece of electronic junk" after your current project is over. Although you might not need to measure forces in all 6 axes right now, a 6-axis FT Sensor is usually a good investment because of its accuracy and flexibility and therefore is a good all-round choice.

However, there may be other specific limits (such as size, cost, working environment, etc) which mean that another type of force sensor might be better suited to your application.

Below we introduce some of the various force sensor types which have been used in robotics research.

TYPES OF FORCE SENSOR

Simple Pressure Sensor

The most basic force sensor is a simple Force Sensitive Resistor. These can be ridiculously cheap (some as low as 5 USD), and you can even make one yourself using some two pieces of wire and some conductive foam. In general, they are not suitable for precision measurements but are good for detecting if a force has been applied or not. The error may be as high as 25%, but if that's good enough for your application then these are certainly an economical option.

Capacitive and Resistive Flexible Force Sensors

A step up from a Force Sensitive Resistor are flexible force sensors. While they may look similar to the previous option, and work on similar principles, they generally have a better accuracy and are also a bit more expensive (more in the range of 20 USD for a resistive flexible sensor). The Flexiforce, for example, is a popular brand of flexible resistive force sensors which can have rated errors of less than 3% and support loads up to 445N. They are adaptable and fit into small spaces, but only detect forces applied at one point and in one direction. Capacitive flexible force sensors are a similar option, which can be a bit more accurate over a smaller range, but are also more expensive.

Piezoelectric

There are quite a large range of piezoelectric force sensors, from the nano scale right up to forces in the kN range. They work using quartz crystals which generate an electrostatic charge the instant a force is applied, creating a voltage which is proportional to the input force. They can be good for dynamic force applications due to their high responsiveness. However, they do have quite a restriction – they are not good for static force measurements. This is because the voltage decays quickly, meaning you would have to employ lot of extra signal processing for use in robotic applications.

Strain Gauge Based

By far the most common force sensing technology in robotics is the strain gauge. These are either used individually, in a single load cell, or multiple strain gauges are used together to measure multi-axis forces, as in a 6-axis FT Sensor. The function of a strain gauge is quite simple – the resistance of a flexible conductive foil changes when placed under strain. This resistance change is usually measured using a Wheatstone Bridge, a highly accurate analog circuit which can be set to produce a voltage proportional to the resistance change.

When used in force sensors, the strain gauge is fixed to a material with known mechanical properties, e.g. steel. When a force is applied to the material it will deform elastically. The strain gauge measures this deformation and (using simple materials theory) this is multiplied by the Young's Modulus and cross sectional area of the material (which are both constants) to calculate the force.

Force sensors of one axis are known as "load cells". You can buy them for a variety of different loads and sensitivities. This list from SensorLand explains the different types of load cell. Of course, unless the load cell has integrated electronics you will have to perform some signal processing on the output of the load cell.

Most of the FT Sensors which are used in robotics already have integrated electronics, which saves you from having to perform this signal processing and ensures that it is carried out as optimally as possible. Many include easy communication options, dedicated software libraries and even integration with common frameworks like ROS, which we'll cover in Part 2.

OTHER SENSOR TECHNOLOGIES

Although we have mentioned some common types here, there are many more technologies which are used in force sensing. These include pressure-based sensing, using pneumatics or hydraulics, which have been used in research for micro applications and can be used in restrictive environments such as fMRI machines. There are also less common techniques, like magnetic force sensors. New force sensors and force sensing technologies are being developed all the time, especially in the field of micro and nano-electronics, as used in robotic micromanipulators. There are even new, cheap, tactile sensors which are based on barometer technologies, which have been integrated into the fingertips of our Robotiq 3-Fingered hand.

A NOTE ON CUTTING EDGE FORCE SENSING TECHNOLOGIES

One of the great things about being a researcher is that you have access to research papers about all the cutting edge technologies. This means that you can read about all the brand new force sensing technologies which are being investigated in research labs right now. However, many of these new technologies are only at the developmental stage and are not available in off-the-shelf products.

Some of the new technologies look to have advantages over existing FT Sensing technologies, like parallel-mechanism force sensors based on the Stewart Platform. As these technologies are still in development, it is likely that you would have to build and implement the entire sensor yourself in order to utilize the technologies.

You will have to decide whether the costs associated with such development would be justified by the benefits of the technology. If not, its probably better to choose a more established technology.

Even if you do decide to implement a new force sensing technology, you may still need a reliable 6-axis FT Sensor to prove or compare its performance in an experimental setting.

WHAT ABOUT TACTILE SENSING?

We can't leave the topic of force sensor types without mentioning tactile sensing.

You might ask: What's the difference between a tactile sensor and a force sensor?

Basically the difference is one of application. Most of the sensing technologies mentioned in this section are used for both force sensing and tactile sensing. Tactile force sensors are much smaller because they are placed on the fingertips or "skin" of a robot. They are also used to measure much lower forces, usually at a higher accuracy.

From a practical perspective, a FT Sensor will usually be integrated into the control loop of the robot arm. A tactile sensor will be integrated into the control loop of the robot's fingers. We have discussed this further on our blog, For an excellent introduction to tactile sensors in robotics, see the Chapter on Force and Tactile Sensing in the Springer Handbook of Robotics.

HOW TO CHOOSE A FT SENSOR

There are many FT Sensors on the market so you might find it difficult to choose between the options.

Just like any other engineering product, it gets easier to choose when you have some past experience with the product. For small products, like microprocessors, it's feasible just to buy a few models and try them out. However, FT Sensors can often be quite pricey so this isn't really an option. You wouldn't want to end up with a sensor which doesn't suit the needs of your application so it's important to get familiar with the specifications before you buy one.

In this section, we'll outline some of the different specifications which are used to describe FT Sensors and which are the most important for robotics research. You will need to balance many different specifications to decide which sensor is the best fit for your application and budget.

WHICH SPECIFICATIONS ARE IMPORTANT TO ROBOTICS?

In general, the following specifications are most important when buying a FT Sensor for robotics. Your application may require specific performance in other areas as well.

You might want to look at the data-sheet for a 6-Axis FT Sensor while you read this section, so you can pick out the specifications. Why not look at the Robotiq FT-300 data-sheet as an example? (just click this link)

FORCE AND TORQUE MEASURING RANGE

This is the range of forces and torques that the sensor can detect. The force measurements (Fx, Fy and Fz) will be rated in Newtons (N) whereas the torque will be rated in Newton-meters (N.m.). Often, FT Sensors can measure an equal range in both directions, which is noted by a +/- sign. This means, for a +/-150 Newton sensor, it can measure 150 N in each direction.

For example, this would mean that the sensor can measure the lift (pull) of an object up to 150 N and it can also measure the applied force (push) of 150 N in the opposite direction. Don't forget about the weight of the object! The weight of anything held by the robot counted towards the total 150 N of lift in any direction.

MAXIMUM OVERLOAD CAPACITY

Every mechanical device has a limit beyond which it will break. For FT Sensors, this is either presented on the data-sheet as a force (e.g. 1 kN for the FT-300) or it is presented as a multiple of the maximum measurement values (e.g. 5 x max measurement or 500% max measurement). It is vital that you know the overload capacity of your FT Sensor!

This value represents the force which will provide permanent damage to the sensor, so you really don't want to be applying forces which are even close to it. This is particularly important if you are mounting heavy mechanical parts onto the sensor for external force data (as described Part 3).

Remember that the maximum force you apply to the sensor is dependent on more than just the static forces. It is important to also consider the dynamic forces which might be applied. For example, if your robot were moving a 10 kg weight at a horizontal constant velocity of 2 m/s, collision with a solid wall could easily produce an impact force of 4 kN or more. If you had only calculated the overload capacity based on the object's static mass under gravity, you'd have only considered a force of 98.01 N which could easily lead to a broken FT Sensor. See this page on The Engineering Toolbox to refresh your memory on how to calculate dynamic impact forces.

Try calculating the maximum dynamic force the robot could apply when holding its maximum payload and colliding with a solid wall at top speed. Probably this will be a higher force than the maximum overload capacity of many FT Sensors. Consider what measures you can take to ensure that no such forces are applied to the sensor.

FREQUENCY/MAXIMUM DATA OUTPUT RATE

The Maximum Data Output Rate defines the maximum sampling frequency of the sensor. This determines how often it can update you with new force data. This value will be mostly defined by the integrated electronics of the sensor. It will also be affected by the communication interface which is used.

High frequencies are important for some robotic applications and less important for others. For example, research into haptic and bilateral teleoperation often require that control loops achieve a frequency of at least 1 kHz. Other robotic applications might need much lower frequencies, especially if they have to perform time-consuming calculations which use the sensor data.

Work out which frequency you require for your application and choose a FT Sensor which can supply at least this frequency.

LATENCY

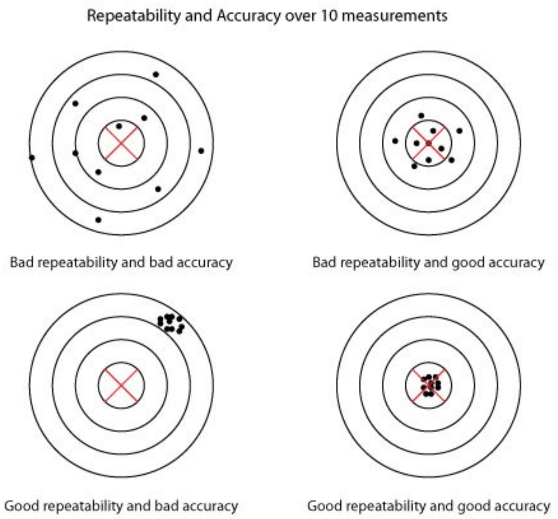
The latency of a sensor is closely related to its frequency. It is measured in seconds (or more likely microseconds). It determines how quickly a change in the input force will be displayed in the data. Lower latency will mean more a responsive sensor. Like frequency, it is also highly affected by the communication options which you choose.

DESCRIBING FT SENSOR PRECISION

There are three different specifications for precision which can be applied to FT Sensors. The most common one to see on data-sheets is the Resolution of the sensor. Resolution, Accuracy and Repeatability are closely related, but they're not the same.

PRECISION – RESOLUTION

Resolution is the minimal change in force that the sensor can measure. Like measuring range, this will usually also be rated in N and N.m. The main thing that limits a sensor's resolution is the sensor noise. A sensor will need a higher resolution (indicated by a smaller number) if the application needs more precision, e.g. when handling very light objects. For rough applications and heavy objects, the resolution can be coarser. Choose a sensor with a resolution which is considerably lower than the smallest measurement you'll want to use.


Repeatability and Accuracy over 10 measurements

Bad repeatability and bad accuracy

Bad repeatability and good accuracy

Good repeatability and bad accuracy

Good repeatability and good accuracy

PRECISION – ACCURACY

Accuracy defines how precisely the FT Sensor can measure the applied force and torque. If the sensor reads 50 N, is the force really 50 N or is it actually 51N? In general, a more accurate force sensor is better. However, like everything else, you don't need to get a super-accurate sensor if it won't make any difference to your application.

PRECISION – REPEATABILITY

The third measure of precision is repeatability. If you apply exactly the same force again and again, will you obtain the same value each time?

It can sometimes be difficult to obtain data about all three of these measures for a particular FT Sensor. At the very least, you should be able to find the resolution, which is probably the most important for robotics applications.

A FINAL NOTE ON CHOOSING SPECIFICATIONS

There are many other specifications which can be used to describe FT Sensors. The specifications listed here are those which are most likely to have an effect on your choice of a FT Sensor for research purposes, and also those most likely to be provided by the manufacturers. If you want to find out more, we have included other specifications in our previous ebook on using FT Sensors in industry. We have also covered specifications like strength, noise level and hysteresis on our blog.

As with many engineering decisions, choosing between different products is a balancing act. Products with better technical specifications almost always cost more money. Therefore, take some time to really work out how what performance you will require of the FT Sensor and purchase one which can supply that performance, or just a little bit higher.

PART 2: USING FORCE SENSORS IN RESEARCH

5 COMMON USES OF THE FT SENSORS IN ROBOTICS

FT Sensors are used for a variety of different purposes in robotics research. Some common uses in robotics research are listed below, but you are only really limited by your imagination and practicality.

1. IMPLEMENTING FORCE CONTROL

Force control is by far the most common application of FT Sensors in manipulator research. The purposes of this type of control include: avoiding unnecessarily high forces being applied to the environment (which can happen with other control methods), allowing more precise manipulation, and coping with deviations from the planned task (e.g. due to manufacturing inaccuracies of a part).

One big advantage is safety. Using the simpler Position Control, for example, you command the manipulator to a particular position. If the robot hits anything on its journey to that position (e.g. a human) it will have no way of knowing how much force it has applied and could cause serious damage.

Incorporating a force sensor allows the robot's controller to include the force data and therefore ensure it does not apply too much force to the environment. For this reason, force sensors are an integral part of active compliance techniques.

See the section below on force control for information on how to get started with this application.

2. FLEXIBLE AND PRECISION ROBOTIC ASSEMBLY

Although force sensing has only really taken off in industry relatively recently, researchers have been trying to solve the difficult problems of flexible robotic assembly for years. FT Sensors have long been used to generate force data for this application. The task of "peg-in-hole" has been extensively used as a benchmark for assembly tasks. Recent developments introduce even more complexity into assembly tasks, such as performing mechanical assembly in space.

3. SURGICAL ROBOTICS

Recently, there has been a huge rise in research studies into robotic surgery. The needs of this application are quite particular, as sensors are required to be very small (on the micro or nano scale) and to operate in very restrictive environments. As a result, many recent studies introduce novel force sensors, ranging from uniaxial flexure-based sensors, optical sensors and even 6-axis FT Sensors. The field of surgical robotics is relatively new. As a result, there is little agreement about the needs of FT Sensors in this application and some researchers have even questioned if force feedback is necessary (and found that, yes, it seems to be beneficial).

4. HUMANOID WALKING AND MANIPULATION

Humanoid robots are a hugely popular research application for force sensors. FT Sensors can be integrated into almost all parts of a humanoid, from the shoulders right down to to the feet. Force data in all six axes is used in the ankle and/or knee joints to allow two legged robots to balance.

Notable humanoids which use multiple FT Sensors include: Honda humanoids, NASA's Robonaut and Valkerie humanoids.

5. PARAMETER ESTIMATION

Robot control methods are often restricted by the fact that they require precise knowledge of the robotic system and its static and dynamic properties. Research into parameter estimation techniques attempts to solve this issue by estimating the robot's parameters based on sensor data. FT Sensors are often used for this application. They can be attached to the base of the robot, onto the wrist or a combination. Parameter estimation techniques can also be used to estimate the properties of manipulated objects.

These are just some popular research applications for FT Sensors. There are many different ways that you can get the most out of an investment in a FT Sensor. Try reading other people's research papers for ideas on how you might be able to use them in your own research.

INTERFACING WITH FT SENSORS

In order to get the most out of your FT Sensor you need to interface with it, both physically and through software. Depending on the type of force sensor you choose, your options will differ greatly in complexity. There are two parts to the interface:

- **THE PHYSICAL INTERFACE** — including mechanical structure, plugs, connectors, electronics, etc.
- **THE SOFTWARE INTERFACE** — where you convert the raw data from the sensor into the required format for your application.

Three key factors which affect the choice of interface are:

TIME AVAILABLE FOR INTEGRATION

Sensors which require you to add low-level electronics (e.g. analog circuits or TTL) will take longer to integrate than sensors which include their own electronics. On top of that, if you want to use force control on your robot you will likely have to implement it yourself. See the section below on force control if you want help on where to start with implementation.

Since very recently, there are also some options if you want to get the advantages of force feedback without having to spend a long time implementing force control yourself. With the recent rise of force control in industry, some manufacturers have started to provide easy-integration options between popular robots and FT Sensors. For example, the FT-300 has been developed to work out-of-the-box with the Universal Robots brand. This is quite a new development in industrial robotics, which is likely to become more common in the coming years. Researchers can certainly take advantage of this trend by choosing FT Sensors which have been “pre-integrated” with their robots.

REQUIRED RELIABILITY OF THE SENSOR

Sensors which work "off-the-shelf" will be more reliable than those that you develop yourself. This is partly be due to lack of resources in most labs. However, it is mostly because the aim of robotics research is usually to build proof-of-concept systems, not highly reliable products. If reliability is important, you should seriously consider this when shopping around for FT Sensors.

REQUIRED PERFORMANCE

As we saw in Part 1 there are several key specifications to take into account when choosing a FT Sensor. Three of these specifications are important when considering interfacing options:

- The achievable sampling frequency from the sensor.
- The achievable latency from the sensor.
- The physical interfacing options that the sensor supports.

Your choice of both electronic and software interface will be highly dependent on these factors.

BEWARE OF “FALSE ECONOMY”

Don't make unnecessary extra work for yourself by choosing a FT Sensor which requires a lot of extra development to integrate. Try to pick the most off-the-shelf technology that makes sense for your application and budget.

Hypothetically, you could develop your own 6-axis FT Sensor out of steel and strain gauges. It would be probably be cheaper than buying a 6-axis FT Sensor, if you only consider material costs. However, this lower price wouldn't include the hugely long development and integration time. The sensor would also be less reliable than an off-the-shelf product.

Unless a brand new sensor is the target output of your research, it usually makes more sense to buy an integrated sensor which makes interfacing easy and devote your precious time to the most important parts of your research.

PHYSICAL INTERFACE

The physical interface with the sensor incorporates both mechanical and electric/electronic interfaces.

MECHANICAL INTERFACE

As described in Part 1, there are basically two options for mechanical fixing: you can fix the sensor to the robot or to an external structure. The actual mechanical fixing will depend on which force sensor you choose and what you want to attach it to. Some force sensor types have particular fixing methods (e.g. Flexiforce sensors are better fixed with tape than glue), so check the manufacturer's instructions and also search online to find out how other people have attached the same sensor.

If you're attaching to the robot directly, find out if the sensor is compatible out of box, or if either manufacturer (of the robot or the sensor) supplies a suitable adapter plate. This makes things a lot simpler, as you can see in this video of our Universal Robot adapter.

Even if there is no adapter plate available from the manufacturer, a good machine shop can usually build a suitable one out of a piece of metal. If you can't even afford this (having spent all of your research budget on the force sensor itself) you could always build one yourself if you're handy with tools. The adapter will usually only consist of a metal plate with countersunk holes to match the holes at the end of the robot arm on one side and the holes of the sensor on the other.

The data-sheet for the FT Sensor will provide a mechanical drawing of the sensor, from which you can design the mechanical connector. Some manufacturers (like us) also provide CAD files which you can use when integrating FT Sensors into larger mechanical designs.

Note: If you build your own adapter plate, make sure that all holes are included in your design and are securely fastened. Failure to attach some of the holes may result in incorrect force readings, which could lead to hours of unnecessary, headache-inducing debugging. Also, ensure that the material you use for the adapter plate can withstand forces up to at least the overload capacity of the sensor.

ELECTRONIC INTERFACE

The electronic interface with your Force sensor is not quite as simple as the mechanical interface. Depending on which force sensor you've chosen, you will have different options.

Force Sensitive Resistors

If you have chosen to use Force Sensitive Resistors, there are quite a few options for interfacing with the electronics. These range from a simple voltage divider to more involved Multi-channel FSR-to-Digital circuits. They will require you to develop analog circuits with Op-Amps. See the Sparkfun FSR Integration Guide for an in-depth guide.

STRAIN GAUGES

In general, as described in Part 1, strain-gauges are conditioned using a Wheatstone Bridge which converts the strain value into a voltage. If you are using a strain-gauge without integrated electronics and need to implement your own signal conditioning circuit, I can recommend this guide from Sensors Mag and this one from Texas Instruments.

6-AXIS FT SENSOR

FT sensors with integrated electronics and communication have the benefit that you don't have to design and build signal conditioning circuits. This is a bonus, as it means that you can be sure the signal conditioning has been carried out as optimally as possible by the manufacturer. However, it also means that the choice of communication protocol is no longer in your hands. You must be able to interface with the device using the communication options chosen by the manufacturer.

Many of the best FT Sensors will give you several options for communication with the sensor, so you can pick the one which suits you most. For example, the Robotiq FT Sensors have options of USB and RS-232/RS-485 connections (although they have multiple other communication options when used through the Universal Controller).

When choosing which method to interface with an FT Sensor, one of the biggest considerations is the sampling frequency that you require and the acceptable latency. Although FT Sensors often have a very high available frequency (some up to 50 kHz), not all interface options can support high speeds. Therefore, it is important to decide on your desired performance and choose an FT Sensor and interface option that can achieve this performance. Higher frequencies are not always better – they may not be necessary if the rest of your system is running at a much lower frequency.

COMMUNICATION OPTIONS FOR FT SENSORS

Some common communication options for FT Sensors are:

USB

USB is possibly the closest thing we have to a standard connector for everything. As a result, it is a pretty safe option when interfacing your FT Sensor with a computer. USB 2.0 consists of four connectors (Vcc, Gnd, Data + and Data -), while the newer USB 3.0 has five additional pins to hugely increase data transfer speeds (Transmit +/-, Receive +/- and an extra Gnd). To use a USB FT Sensor, you only need the correct driver software installed on the computer and everything should work out of the box (if it doesn't, first try the forums then contact the manufacturer directly). If you want to interface with low-level TTL or various Serial protocols, you can convert the signal using the well known FTDI chips.

RS-232/RS-485

These serial interfaces have been being used since right at the beginning of computing (1962 in fact). As a result, they are very well supported and, before USB, were "the standard serial protocol" for computing. Both types have connectors with 9 pins, although in practise often only four of these pins are used (Signal, Ground, Transmit and Receive). Given the simplicity of the protocol, it is often possible to integrate directly into electronics using the corresponding chips. There are also many off-the-shelf converters to convert to USB, Ethernet and other common connectors without having to solder anything.

For a good introduction to RS-232 and RS-485 see this web page.

ETHERNET OR ETHERNET/IP

Ethernet is also a popular interface protocol for some FT Sensors, due to the ease of integration into networks - you just have to plug them into a router. Often, sensors have the option of using Ethernet/IP protocols, such as EtherCAT. We have discussed the difference between Ethernet and Ethernet/IP on our blog (IP stands for Industrial Protocol). They use the same physical Ethernet cable, but Ethernet/IP options allow easier integration with existing systems. These options can affect the achievable frequency and latency and may require special hardware (such as our Universal Controller).

OTHER PROTOCOLS

There are other various protocols, such as the Serial I2C protocol (also called Two-Wire), CAN, UART, SPI and others. Not many 6-axis FT Sensors have these options, but some of them do. If you decide on an FT Sensor with these interface options, make sure that the protocol can handle your desired sampling frequency and latency as they can have low data rates.

SOFTWARE INTERFACE

After you have connected your FT Sensor, you need to be able to use the data in your application. This means accessing the data in code, which will largely depend on which programming language(s) you're already using. However, there are some considerations to make when choosing how to implement your software:

WHAT LIBRARIES ARE AVAILABLE FOR THE FT SENSOR

You need to decide if you want to use the libraries provided with the hardware or if you want to write your own code from scratch. It is usually (but not always) a good idea to use the manufacturer's library because it has been programmed by people who know the idiosyncrasies of the device very well. By using the provided library, you're usually given a very clean interface to work with. However, this does mean that you'll have to use the programming languages chosen by the manufacturer.

C/C++ is a common language for FT Sensor libraries. If you want to use another language, you might have to write a wrapper library or find one already written by somebody else.

Very occasionally you will find a software library which is so badly written that it's easier to throw it away and code your own. However, before you do this, search thoroughly to make sure that nobody else has already written an alternative library before trying to code your own. If you've had a problem with the provided library, chances are someone else has also had the same problems.

At Robotiq, we have worked to make our FT sensor library as clear and straightforward as possible. You can download it from the support section of our website. If you are having any difficulties with the library, don't hesitate to contact us and we'll be happy to help.

DO YOU REQUIRE REAL-TIME PERFORMANCE?

The data from FT Sensors are often used inside the real-time control loops. Obviously, this means that the sensor must be able to provide real-time performance, which most of them do. However, some programming languages or systems (including ROS in some configurations, see Part 3) do not have the capability to keep-up with real-time demands. There are ways of getting around these limitations, but it's something that you'll have to think about right from the beginning.

WILL YOU WANT TO REUSE YOUR CODE IN THE FUTURE?

Finally, there is the question of re-usability. As researchers, it can often take us a long time to develop a particular software system. It makes sense to code the software in a way which is easily reusable by other researchers. However, often research code is messy due to lack of time and strict coding practices, as exist in industry.

Think about whether it is likely that anyone (including yourself) will want to further develop your code in the future. If so, consider a coding environment which makes this easy. One of the biggest reasons that ROS (see Part 3) has become so popular in recent years is because it makes sharing research code very easy.

GETTING STARTED WITH FORCE CONTROL

One of the most common applications of force sensing in research is to implement Force Control. Since the 1970s, researchers have developed a whole load of novel ways to integrate the sense of touch into their robots.

Force control methods are usually compared to motion control methods, such as position control and velocity control. If you are interested in learning more about motion control, Chapter 6 of the Springer Handbook of Robotics gives a good overview.

The most basic type of motion control is pure position control, whilst the most basic type of force control is pure force control. Here are the differences between the two:

PURE POSITION CONTROL	**PURE FORCE CONTROL**
● You command the robot's joint to a particular position.	● You command the robot to apply a particular force to the environment.
● The robot tries to reach that point.	● It will try to apply that force.
● If it cannot reach the point (e.g. if it collides with something) it will continue to apply a stronger and stronger force on the environment until it does.	● If the robot does not apply a force (e.g. when moving in free space) it will keep moving in the same direction until the target force is reached.

Clearly, both of these control systems have some disadvantages. Pure position control has the potential to cause real damage to the environment by applying unregulated forces. On the other hand, pure force control makes moving the robot precisely through free space quite a tricky task.

In practice, force control is usually integrated into robots by using a combination of motion and force control.

HOW TO START SMALL

Before you begin, consider if it would be better to start small.

It is quite common practice for new researchers to implement control systems on a 1-DoF (Degree of Freedom) robot. This “robot” usually consists of a stick (representing the robot's link) attached to a motor or other actuator.

As a development task, this can be quite a boring thing to do. However, it serves a very important purpose when trying out new control systems – it allows you to solve the issues of the controller on a small scale before moving onto a larger robot.

If you are using a 6-DoF robot, any errors in your implementation of force control will be magnified by 6. It will be difficult to ascertain exactly where the problem is. Therefore, to avoid giving yourself a headache, try implementing force control on, say, only one joint of the robot and apply the control to the rest of the robot only when you know it's ready.

TYPES OF FORCE CONTROL

There are several types of force control which combine the benefits of both motion and force control approaches.

These are all "active" force controllers (i.e. Where force data is incorporated into the control loops). You can also have "passive" force control, such as when a bendy material or springs are integrated into a manipulator to ensure that it cannot apply a high force.

INDIRECT FORCE CONTROL

These methods are primarily motion controllers, which only apply a force constraint when the position of the robot deviates from the target position. These controllers do not explicitly close the force-feedback loop. This means that, technically, they don't need to be implemented using FT Sensors However, often it is a good idea to use the data from a FT Sensor data improve the controller performance.

- **IMPEDANCE CONTROL —** This applies a mass-spring-damper between the target position and the actual position of the robot. Imagine that you attach a spring between your finger and some target position in the air. The further you move your finger away from that point, the more force the spring applies to pull you back to that point. This video shows what impedance control looks like on a robot.

- **ADMITTANCE CONTROL —** This is the inverse of impedance control (more or less). It can be better imagined as pushing your finger through a very viscous substance, like honey or wet sand. The more force you apply to the substance, the further your finger will move. This video shows what admittance control looks like on a robot.

There are also special cases of impedance and admittance control, such as stiffness control which actively varies the stiffness of the robot joint (demonstrated quite clearly in this video).

DIRECT FORCE CONTROL

These methods incorporate both force and motion into the control of the robot. They require two different inputs - the target position/motion and the target force. As a result of this, some direct force controllers rely on quite a clear model of the task.

This can sometimes be restrictive when you are using the controller in unknown environments or for unknown tasks. However, when you do know explicit details about the task, these controllers allow you to independently control the each axis in either force or motion depending on the needs of the task.

- **HYBRID FORCE/POSITION CONTROL —** This control method is quite a popular one. It separates all 6 axes of the task (3 force and 3 torque) and applies either a motion based control or a force based control onto each of the axes. Unconstrained (free) axes are controlled in position while constrained axes are controlled by applying a constant force. This video shows what hybrid force/motion control looks like on a robot.

- **PARALLEL FORCE/POSITION CONTROL —** This is an option if you want to use direct control but do not know details of the task. It works by implementing both force control and position control simultaneously. The controller is set up so that the force control has more effect on the output than the position control does. This means that the robot will move under the position control until an external force is applied.

These are the most common types of force control and they all can all be applied using 6-Axis FT Sensors. Depending on your application, you might choose to implement one of them or even use several controllers and switch between them for different tasks.

WHERE TO LEARN MORE ABOUT IMPLEMENTING FORCE CONTROL

This section has was a quick introduction to force control. Obviously, you will need more information if you want to actually implement these controllers using your own robot and FT Sensor.

We've collated a free list of great resources where you can find detailed information to get started implementing force control.

If you downloaded this ebook from the Robotiq website, the free list of resources should be making its way to your email inbox any time soon.

However, if you found this ebook elsewhere online, never fear. You have not missed out! Jump to Part 4 of this ebook to find out how you can get the free resources.

PART 3: DESIGNING EXPERIMENTS WITH FORCE

HOW TO INTEGRATE A FORCE SENSOR INTO A ROBOTICS EXPERIMENT

Experimental design is a hugely important part of robotics research. We have introduced the fundamentals of good experimental design in one of our popular blog posts.

From the perspective of experimental design, FT Sensors can be used in robotics research in (at least) two different ways:

- **FT SENSOR(S) ARE ONLY USED IN THE ROBOT'S CONTROL SYSTEMS —** In this case, the FT Sensor is a vital part of the robot control itself; the robot would not function fully without the sensor.
- **FT SENSOR(S) ARE ONLY USED TO GENERATE DATA FOR THE EXPERIMENT —** The robotic system would function perfectly without the FT Sensor but you are using the sensor to collect data.
- **FT SENSOR(S) ARE USED IN BOTH THE ROBOT CONTROLLER AND FOR DATA —** The robot's functionality relies on the force sensor data, but you are also collecting that data for the experiment.

In this section, we'll focus on the second two of these ways – using 6-axis FT Sensors to generate data for research experiments.

WHEN TO USE FT SENSORS IN AN EXPERIMENT

As we discussed in Part 2, there are many applications of force sensors in robotics research. Many of these use force data as an integral part of the robot's control systems. However, you can also use force data to generate experimental data.

Some examples of this could be:

- Proving that the robot applies minimal manipulation forces to the environment or task.
- Detecting touch/impact events to isolate parts of the task.
- Performing pre-manipulation calibration of objects by weight.
- Verification of the performance of a novel force sensor.
- As a touch input for human interaction studies.

You should use a 6-axis FT Sensor for any experiment or system which requires reliable, high quality data about physical interactions. These interactions could be either within the robot's structure (e.g. in a shoulder joint) or between the robot and the environment.

WHEN NOT TO USE FT SENSORS IN RESEARCH

It's important to choose a sensor only if it really makes sense for your experiment. This means that all the collected data should be related to your experiment's hypothesis (or hypotheses). For an introduction to hypotheses, see our blog post on good experimental design practices for robotics research.

Basically, if you don't need force data don't include a force sensor.

Feel free to log the data if you already happen to have a force sensor attached to the robot or task. However, if your experiment would be better and clearer by using another piece of data (e.g. speed or acceleration) then don't include the force just because you've got a handy FT Sensor.

This might seem obvious, but it can easily be forgotten when you are in the stressful situation of having an experiment which is not producing exciting data. It can sometimes be tempting to just add a new sensor and see if the data it generates looks more interesting. However, this approach rarely works to magically make your experiment publishable. When in doubt, simplify your experiment or reassess the hypothesis.

WHERE TO PUT THE FT SENSOR

As discussed, there are two main positions you can place an FT Sensor: Attached to the robot or attached to something else. Both of these have advantages and disadvantages.

On-Robot

By far the most common situation is that an FT Sensor is placed at the end of a robot arm. This is usually because the data is integrated into the control systems of the robot to introduce force control (see Part 2 of this ebook as well as our blog posts for an introduction to force control and some of the challenges for integration).

ADVANTAGES	DISADVANTAGES
● Can use the force data to implement force control.	● More complex equations required to convert forces in the global coordinate system for analysis.
● Detect the force applied by the robot in any area of the workspace.	● Can be complicated to differentiate specific touch events from the rest of the data.
● Widely recognized as a standard method of	

using force sensors in robotics research.

- Power and communication cabling must be attached to the robot itself.

Off-Robot

The alternative is to place the force sensor onto the workspace itself. This can be a very quick and robust way of obtaining experimental data if you can attach the FT Sensor directly to a test setup (e.g. as is done by Hannaford et al 2002).

If you are using the FT Sensor as part of an input device, then it is more likely that it will be fixed in a static position (as you can see in this video where a joystick is added). In this case, it would not interact with the robot but rather the human operator.

ADVANTAGES

- Very quick to implement.
- Force data is already in the global coordinate system or requires little conversion.
- No special cabling setup is required.

DISADVANTAGES

- Forces can only be detected in one area of the workspace.
- Care must be taken not to overload the sensor if it is attached to heavy mechanical parts.
- Potential for unbalanced static forces if the test setup is heavier on one side.

USING FORCE SENSING IN ROS

ROS (Robot Operating System) has become hugely popular within the robotics research community. In this section we'll look at why you might use ROS with your FT Sensor.

We'll also introduce the ROS library for the Robotiq FT Sensors, which will allow you to integrate your sensor quickly into your research application. Even if you are using a different FT Sensor, this section will give you a good introduction to using your chosen sensor with ROS.

WHAT IS ROS AND WHY WOULD I WANT IT?

If you're already a “ROS-wizard,” skip this part and jump to the next page where we'll discuss the pros and cons of force sensing in ROS.

Robot Operating System (ROS) has become the closest thing that the robotics community has to a standardized platform. Basically, it is a collection of open-source software tools which provide a communication framework (and more funky functionality) for passing data within robotic systems. It primarily runs on Linux but has now also been implemented on many other platforms.

ROS is (now) hugely powerful because so many people in research are using it. As with most open source initiatives, the bigger the size of the community the more the software can achieve. This allows you to easily use many other people's research code in your own projects.

In the past, using someone else's research code would be a huge pain in the diodes. You would often have to spend many hours reading pages of badly documented code just to understand how it worked. That was if you could even find a copy of the code in the first place! Then, to use the code yourself, you would usually have to rewrite it, test it, debug it and easily suffer a nervous-breakdown in the process. It was a huge waste of time.

Although ROS hasn't solved this problem completely, it has made things a lot easier. If you want to use ROS, you have to code things "the ROS way," which is easily learnt in a few hours using the clear tutorials on the ROS website. This forces researchers to code their algorithms in a way that is inherently reusable. Even if the underlying code is still messy and unreadable, the "outward facing" parts of the code will always consist of simple ROS nodes, messages and services.

For FT Sensors, this means that you don't need to understand exactly how the communication protocols work between the computer and the sensor. You only have to run the program, which will then deal with all the low-level stuff and send ROS a nice clear stream of force data for you to use in your applications.

HOW TO GET STARTED WITH ROS

Getting started with ROS is really easy. You just have to go to wiki.ros.org and follow the tutorials, which show you how to install and use ROS.

If you don't know anything about ROS it is highly recommended to go through the tutorials, as there are some key concepts to learn (such as publishers/subscribers, roscore, etc) which will make no sense to you if you haven't learned them.

If it has been a long time since you last used ROS, it's a good idea to read through the updated tutorials. There are often quite big changes between ROS Distributions, although the core concepts remain the same.

The tutorials are really straightforward and you should be able to get through them pretty easily. In fact, most of the issues which cause new users to think "ROS is difficult" are not actually problems with ROS at all. Instead issues often arise when someone is using a Linux/Unix system for the first time and has only previously used Windows or Mac. ROS runs most easily on Ubuntu Linux, so it makes sense to get comfortable with it.

ROS also requires that you use the Linux command line, which some new researchers have never used before. Here's a short, complete beginner's guide to Linux, a complete beginners guide to Ubuntu and a simple introduction to the Linux command line.

But the first stop for anyone starting out with ROS is its website: www.ros.org

THE PROS AND CONS OF USING FT SENSORS WITH ROS

There are some advantages and disadvantages of using ROS to interface with your FT Sensor. Here are a few you might like to think about:

PROS

- Allows you to easily use more advanced FT Sensor functionalities using existing ROS libraries.
- Simple to integrate FT data into larger robotic projects.
- Eases the integration with many industrial robots using the **ros-industrial** packages.
- Quick to start using the FT data - just run the ROS node and you can read the sensor.
- Does not require any knowledge of the underlying protocol for the sensor.
- Easy to switch to a different FT Sensor later as separates the sensor from other algorithms.

CONS

- Might be too much processing overhead if you don't need other ROS functionality.
- Requires Linux to be running on at least one computer in the network (the ros master needs to be running on Linux, though now it can communicate with Windows)
- ROS is not really an option for hard real-time systems (though there are now some good approaches for using it with real-time elements).
- Not a great option for resource-limited embedded applications.
- Many research labs don't use ROS. Integrating your code with the code of your fellow researchers is sometimes more important than integrating with the wider research community.

A NOTE ON THE USEFULNESS OF ROS REPOSITORIES

You might decide that ROS isn't a good option for your research development. This is fine, there is no need to incorporate it if it doesn't make sense. However, even if you're not going to use it, learning the basics of ROS can be really useful for other code development.

How? Because understanding ROS still means you can integrate other people's research code more easily.

There are hundreds of great software repositories available in ROS, many of which contain the actual code which was used to generate results in research papers. The strict ROS conventions mean that it is a lot easier to read and understand code written by other people. It allows you to "fast-track" your reading of the code and only extract the parts which are important to your application.

By choosing not to learn ROS, you might be missing out on free software libraries that could boost your research to the next level, or save you loads of time by avoiding "reinventing the wheel."

FOUR FUNKY ROS PACKAGES FOR FT SENSORS

There are some great ROS packages out there force sensing. It doesn't matter which sensor you use, most 6-axis FT Sensors can be easily linked to existing code.

robotiq_force_torque_sensor

Of course, we have to mention our own ROS package for the Robotiq 6-axis FT Sensors. This package wraps the communication interface for the sensor into a very easy-to-use ROS node. We introduce some of the specifics of this package below in the next section, but you can also find a guide to using the package in our Online User Manual.

force_torque_tools

This package from KTH Royal Institute Of Technology in Sweden is a good first stop if you want to incorporate your sensor into a robotic manipulator. It provides calibration tools for sensors attached between the arm and end effector. It can calculate: FT Sensor biases, gripper mass and Center of Mass for the gripper. It can also perform gravity compensation. The package accepts its FT values as a geometry_msgs/WrenchStamped.

ee_cart_imped

It can be very informative to see how others have implemented complex controls in ROS before you try on your own robot. This stack has not been updated for a while at time of writing. However, it shows how you can implement stiffness and impedance control for robotic manipulators using ROS Action Servers. Check out the tutorial page, which uses the PR2 model as an example.

walk_tools

If you want to make a walking bipedal robot, one common way to ensure that it remains upright is to calculate a Zero Moment Point. This package from Thomas Moulard takes two force sensors (left and right foot) and calculates the Zero Moment Point as a vector. It has not been updated for a while, but is a good place to start if you're trying to implement something similar.

THE ROBOTIQ FORCE SENSOR ROS PACKAGE

The Robotiq Force Sensor package is (if we can say so ourselves) a great example of how FT Sensors can be easily integrated into ROS. Basically, it is a ROS wrapper for the C driver library. Although the library itself is actually quite simple, the ROS package makes its use even more straightforward.

The **robotiq_force_torque_sensor** package provides the following nodes:

- **rq_sensor —** This node provides all of the code to interact with the sensor. It provides a publisher for the FT data and a service to interact with the other functionalities of the sensor.
- **rq_test_sensor —** This node interacts with the **rq_sensor** node. It listens for the force data and prints them out as a ros_info log message.

The package needs nothing special in the way of dependencies (just the basic ROS ones) and is self contained.

rq_sensor Node

The rq_sensor node provides two ways of getting data out of the sensor. These are:

ROS Topic: *robotiq_force_torque_sensor*

This publishes a custom message called ft_sensor.msg which contains the FT data in the following format:

float32 Fx, float32 Fy, float32 Fz, float32 Mx, float32 My, float32 Mz.

ROS Service: *robotiq_force_torque_sensor_acc*

This service accepts a custom service type called sensor_accessor.srv in the following format:

string command --- string res

The command request accepts the following values:

GET SNU	returns the serial number as string formatted int8 array
GET FWV	returns the firmware version as string formatted int8 array
GET PYE	returns the production year as string formatted int8 array
SET ZRO	This calibrates the sensor's current readings to zero then returns "Done"

LOOKING "UNDER THE HOOD" - WHAT THE CODE IS DOING

Note, this section is only for people who really want to get stuck into the code or want to use the library without ROS.

The code is fairly straightforward. If you want to look around to see how it works, I'd recommend having a look in the code **rq_sensor.cpp**, which provides the publisher and service.

These files are also marked up using doxygen, so if you want a nice clean "user manual", feel free to generate a doxygen from the **src** and **include** folders.

Basically, the main functionality is taken up by the function **rq_sensor_state**. This holds a switch case which acts as a state machine for the 4 start-up states of the sensor. Most of the time the sensor will be in the **RQ_STATE_RUN** state, which listens to the stream and deals with the output. If the stream is lost, it will close the connection and jump back to the **RQ_STATE_INIT** state, which means that if you accidentally unplug the sensor, the code should try to pick it right back up again when you plug it back in.

Communication functionality is separated into the **rq_sensor_com.cpp/h** files, while the **rq_sensor_state.cpp/h** files add the state machine functionality.

The Maximum Frequency of the Code and Sensor

There is a hard-coded 4 milliseconds delay before the stream data is read. This imposes a maximum frequency on the sensor readings by the library.

The maximum rated frequency of the Robotiq FT-150 and FT-300 sensors is 100 Hz so if you're planning on using it to close a 1kHz haptic control loop (for example) you might want to look for another sensor.

Whatever frequency you choose, it is always a good practice to check the publishing frequency of the topic using the tool **rostopic hz** ***/topic_name***, which detects the actual publishing frequency of a topic. If you are using the **rq_sensor** package, this command would be **rostopic hz /robotiq_force_torque_sensor**. This function will listen to the topic for a short while and let you know the actual frequency of publishing.

PRESENTING FORCE DATA IN RESEARCH RESULTS

Graphical images are one of the most important parts of any research publication. Although they might not admit it, many researchers will scan through a paper before reading it and “look at the pictures” to decide if they want to read it in more detail.

It doesn't matter if you are an R&D engineer writing a report or you are an academic researcher presenting your results for a publication, making good graphs is vital if you want people to read your results.

Graphs, figures and tables are all good ways that you can present your data so that your research results can be easily understood. However, they are only useful if they are presented clearly.

THE CHALLENGES OF PRESENTING FORCE DATA

Force data can be very tricky to present clearly.

Some of the challenges of making clear force graphs are:

- 6-Axis FT Sensors have 6 different streams of data. A simple line graph with 6 different lines is already complicated.
- Raw force data can sometimes look quite noisy, even when it isn't.
- Graphical representations of tactile information are not as intuitive to read as other information is.
- FT Sensors often produce a lot of data. Choosing which data to present is not always obvious.

- There are many different ways that you can present the same data.

The best way to learn how to properly present force data in your publications is to read other people's publications and see how they have done it.

We've collated a whole list of freely downloadable academic papers which demonstrate how you can present force data in ways that are both clear and interesting.

If you downloaded this ebook from the Robotiq website, the free list of resources should be making its way to your email inbox any time soon.

However, if you found this ebook elsewhere online, you can still get access to the list. Part 4 of this ebook explains how you can get these free resources.

QUESTIONS TO ASK YOURSELF

You can learn a lot about how to present force data by reading through other people's research papers or reports. While you read them, ask yourself questions about how they have presented the force data. Some example questions are:

- Can I clearly understand what this graph is showing without having to read the text?
- How long did it take me to fully understand the graph?
- Is it clear which parts of the graph represent force data and which represent other data?
- If the graph is intended to show an event (e.g. a touch event), how has the author indicated where the event occurs in the data?
- Have the authors used any “techniques” to make the data clearer (e.g. varying the line style or color to differentiate different data, aligning several separate graphs together on the page to show that they are related, etc)?
- What would I do differently if I was presenting the same data?

You can ask these sort of questions every time you see force data presented. It may be in a presentation, a paper, online or in a report. Asking yourself questions like this will help you to develop an intuitive understanding of what works and what doesn't. You can then use this knowledge to present the data from your own FT Sensor in the best way possible.

PART 4: WHERE TO FIND OUT

HOW TO GET YOUR FREE RESOURCES EMAILS

This ebook is accompanied by a free “mini-series” of emails packed full of great resources about using force sensors in research.

If you downloaded this ebook from the Robotiq website then the emails should be coming to your inbox any time soon.

However, you might found this ebook elsewhere online, or were passed it by a friend. If so, don't worry. You haven't missed out!

Just follow the link below and enter your details to receive your free resource emails.

Download Link: Force Sensors in Robotics Ebook and Email Series

WHAT'S IN THE FREE, 4-PART EMAIL SERIES

Here's what you'll get in these resource emails. They are packed full of links to great resources where you can learn more about force sensors and their applications in robotics research.

BUYERS RESOURCES – HOW TO BUY A FORCE SENSOR FOR ROBOTICS RESEARCH

We know how difficult it can be to shop around the market for a new 6-Axis FT Sensor. There are so many options! In this email we have done the hard work for you. We've gathered information about some of the top force sensors and presented it for you in an easy to read chart.

LEARNERS RESOURCES – WHERE TO LEARN MORE ABOUT FORCE SENSORS IN ROBOTICS

Research is all about finding the right academic publications. We've searched through the literature and collected some great resources about force sensing in robotics. This includes papers about implementing force control, good resources to learn the basics and starting points for your own literature review into more complex force-based techniques.

TROUBLESHOOTING RESOURCES – HOW TO TROUBLESHOOT A FORCE SENSOR

It's always good to know where to start when something goes wrong. This email contains links to some great places to go online when you've got an issue with your force sensor.

PUBLICATION RESOURCES – HOW TO PRESENT FORCE DATA IN RESEARCH PUBLICATIONS

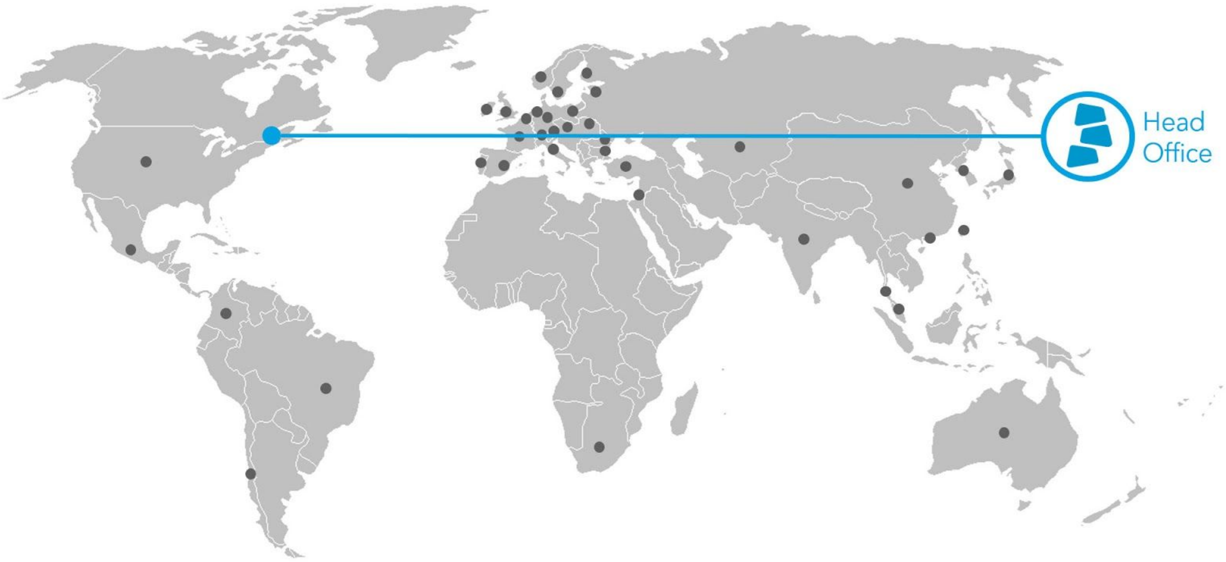
For publication or report-writing, the most important part of using force sensor data is to analyze and present it clearly. This email gives some links to research publications which demonstrate how you can use force data in your own publications.

WHO WE ARE?

Robotiq's Lean Robotics methodology and products enable manufacturers to deploy productive robot cells across their factory.

They leverage the Lean Robotics methodology for faster time to production and increased productivity from their robots. Production engineers standardize on Robotiq's Plug + Play Components for their ease of programming, built-in integration, and adaptability to many processes. They rely on Flow's software suite to accelerate robot projects and optimize robot performance once in production.

Robotiq is the humans behind the robots: an employee-owned business with a passionate team and an international partner network.



LETS KEEP IN TOUCH

For any questions concerning force sensing or if you want to find out how you could integrate our force sensors into your research, contact us.

Workfloor: Robotiq's Blog	Twitter
Linkedin	Facebook
Youtube	Google+

D O F

Robotiq's community where industrial **automation Pros**
share their **know-how** and **get answers**

Ask Your Question

LEARN MORE

robotiq.com | leanrobotics.org