

The ADXL321 is a 2-axis, analog-output accelerometer from Analog Device. The output signal is amplified using an LMC6482 amplifier before sending it to the microcontroller's ADC ports. Digital temperature sensors are interconnected with the microcontroller using I2C bus. They are MCP9803 from Microchip and TMP175 from Texas Instruments, and they make up the on-board distributed temperature network.

Regarding the heating function, the choice went toward insulated etched foil heaters, like MINCO ThermofoilTM or RICA Kapton heaters, which are flexible, readily available and already used as components for space applications, whenever there are constraints regarding weight, low profile, high energy efficiency, as well as precision conductive heat input, vacuum tolerance, and oil contamination.

There are 4 heaters implemented in the smart skin. Each of them is characterized by the following:

- Substrate material: Kapton
- Resistive trace material: Inconel (Ni-Cr)
- Thickness: 50 μm
- Resistance value: 150 \div 300 Ohm
- Nominal operating voltage: 28 V DC
- Maximum operating voltage: 35 V DC
- Power density: $\sim 0.32 \text{ W/cm}^2$
- Total power: 4 W per unit

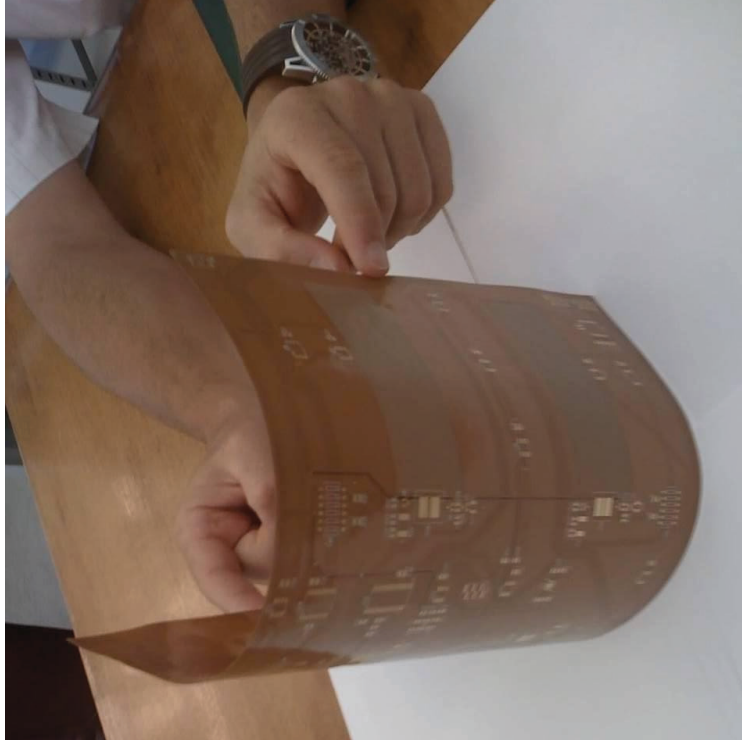
Each heater is also monitored by one MCP9803 temperature sensors.

Due to limitations in the manufacturing capabilities of the smart skin supplier, it was not possible to embed Inconel heaters in the main PCB stack with a single production step, therefore a footprint was provided for SMD heater integration. A modified version of space qualified foil heaters was defined in collaboration with the supplier (RICA): this version is built without one of the two skins, to allow sandwiching with the smart skin's PI substrate.

Looking at the rectangular smart skin board, in its middle zone, both on the right and on the left sides, there are the so called dummy unit areas. These areas represent places where external components (bulky units, payload electronics...) could be installed. In these areas the smart skin is equipped with temperature sensors, so that it can monitor the payload's thermal conditions and, if necessary, act upon it by means of surrounding heaters.

The board is also equipped with RS232, in order to provide a quick, inexpensive, and standard access to verify data transmission during the development process. With an RS232 controller, it is possible to connect directly to a personal computer's serial port and verify the functionality of the board, with particular focus on the communication between on-board microcontroller and slave devices.

The manufacturing process of the demonstrator took advantage of four preliminary steps. At first an evaluation breadboard, made with standard PCB prototyping



(b)



(a)

Figure 2.34: Smart skin: flexible PCB prior to components integration.

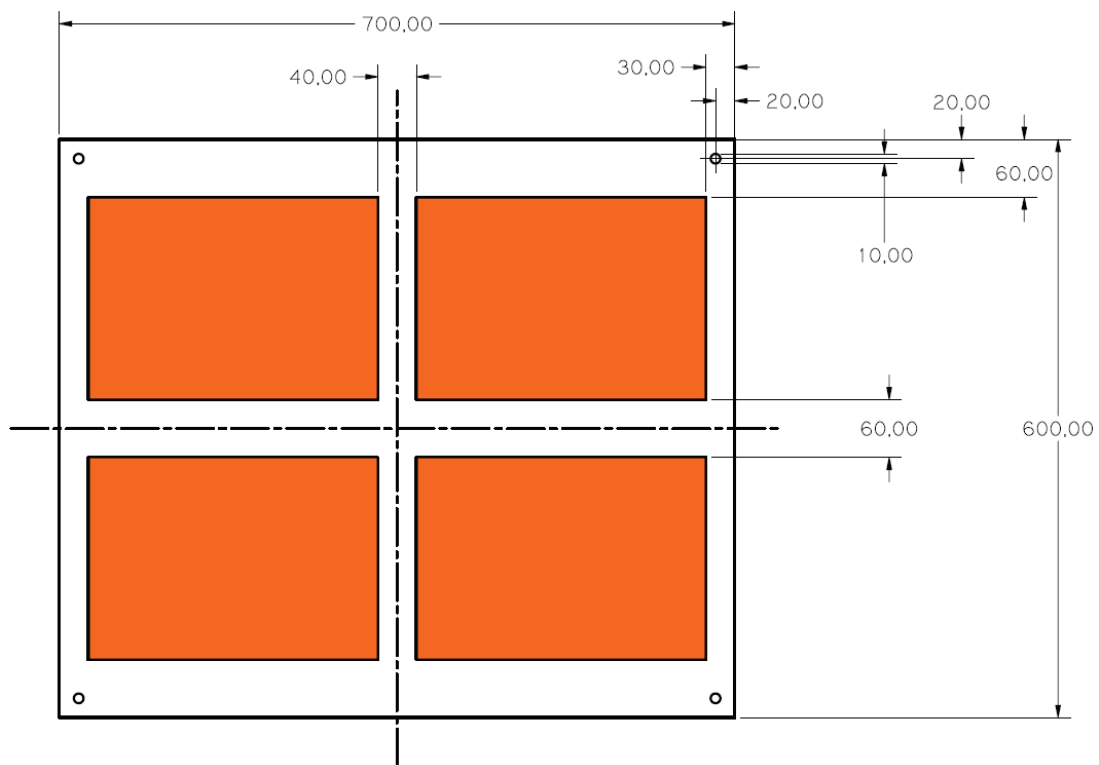


Figure 2.35: Dimensional drawing of the SDA panel.

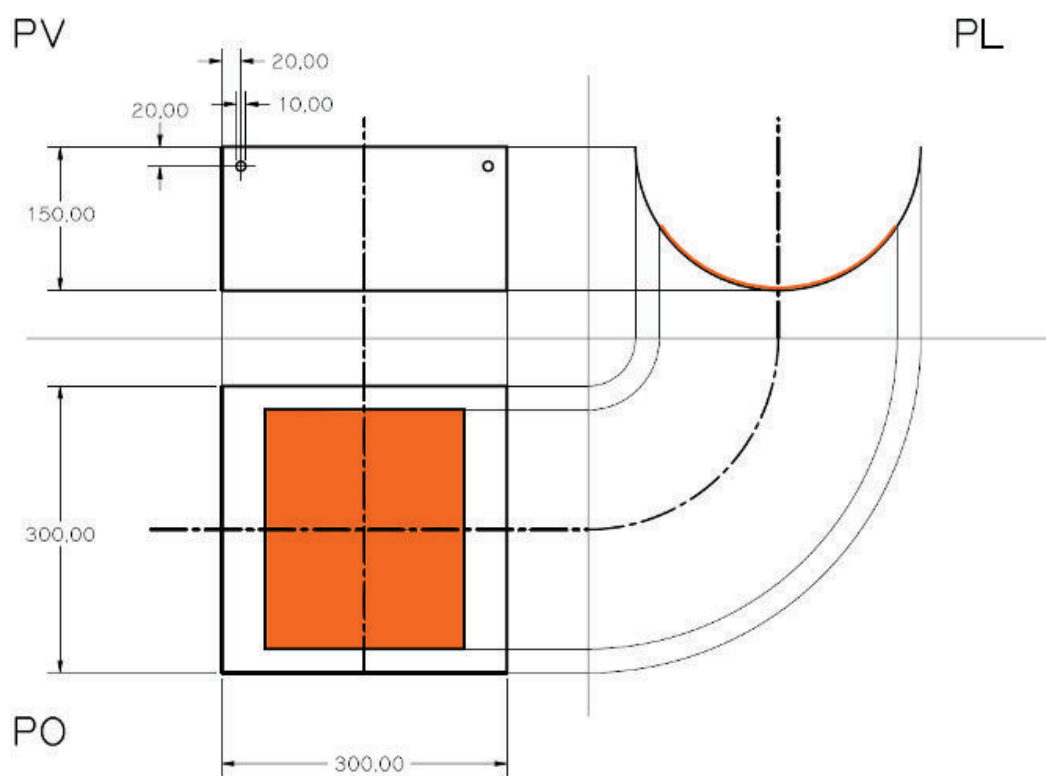


Figure 2.36: Dimensional drawing of the SDB panel, orthogonal projections.

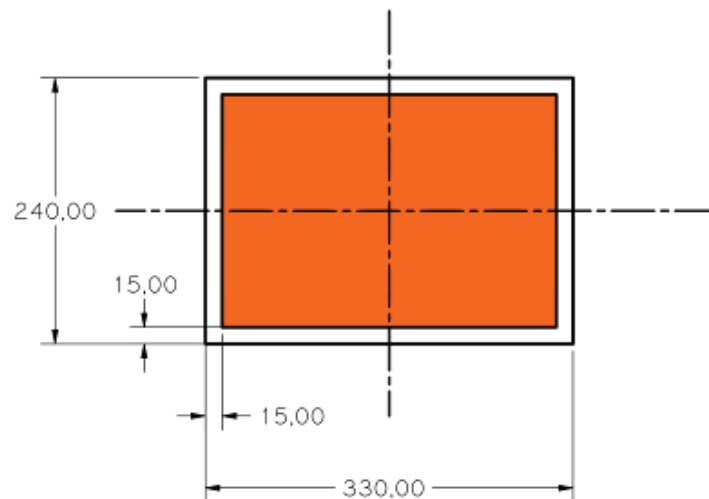


Figure 2.37: Dimensional drawing of the SDC panel.

techniques, was put together in order to verify the preliminary schematics. This board was a proof of concept equipped with just the minimum subset of smart skin components sufficient to evaluate all the functionalities (a microcontroller, a DC/DC converter, one power resistor, power logic to control the heater's dissipated power, four temperature sensors, one inertial sensor, I2C bus, CAN bus interface, and RS232 interface). After successful validation of the preliminary electrical design, this board was also used for initial software development activity.

Then a first prototype, called Proto-A was built. This was a full scale version of the smart skin, and was used to evaluate the electrical architecture, steer an electrical re-design to correct gerber files and bills of materials, and of course develop software.

Other two prototypes were built, Proto-B and Proto-C. Both of them were exploited for hardware debug and software development purposes.

At the end, six samples of the final smart skin design have been manufactured.

Concerning the selection of a structural support panel for the demonstrators, after the test campaign on ABB thermo-structural material, given the fact that the mechanical performance of the Carbon-Carbon sandwich was not fully satisfactory, and given the impossibility to evaluate a new Carbon-Carbon junction technology in a short period of time, a conventional back up solution was selected. The smart skin samples were mounted on classical Aluminium sandwich panels, Aluminium plates, and Kevlar cloth to build, respectively, the demonstrators named SDA (STEPS Demonstrator A), SDB (STEPS Demonstrator B), and SDC (STEPS Demonstrator C).

The first test article, SDA, consists of a 700 mm x 600 mm rectangular sandwich panel equipped with 4 thin, flat, flexible motherboards bonded face to face on one skin. The MBs were glued to the substrate thanks to Hysol® EA9321, a two-component epoxy paste adhesive. The orange squares in Figure 2.35 are the four motherboards, which are identical in functions and components.

The second test article, SDB, consists of a 470 mm x 300 mm shell, which has been curved to obtain the final cylindrical breadboard as reported in Figure 2.36.

This panel is equipped with just one flex motherboard, which is totally identical to those described for SDA. Similarly to what was done for SDA, the SDB motherboard is glued to the substrate thanks to epoxy glue Hysol® EA9321.

As can be seen in Figure 2.37, the third test article, SDC, consists of a 240 mm x 330 mm Kevlar sheet, which has been equipped with just one flex motherboard. The MB is totally identical to those described for SDA and SDB. The smart skin has been bonded to the fabric via 3M Y966 adhesive transfer tape.

Software

Under a software point of view, the microcontroller's firmware needed to read data coming from the sensors and control the independent activation of the four heaters via the corresponding switches, on the basis of trigger events and given control laws. The general architecture foresaw a two level approach: a certain number of given control laws are hard coded in the firmware, but the user can change the main parameters of these laws through an external Graphical User Interface (GUI, also called Human Machine Interface - HMI). For an easy understanding of the software architecture scheme, a general description of the implemented software components is reported below.

Software developed for the smart skin included:

- Microcontroller peripheral drivers. These are: RS232 interface driver, CAN interface driver, I2C interface driver, analog to digital converter driver.
- Timer Interrupts. In order to implement a Real Time Operating System (RTOS), it was necessary to implement a microcontroller timer manager using microcontroller's Interrupt Service Routine (ISR). Two timer interrupts have been used. The first (#INT-TIMER0) is used for application scheduler management, the second (#INT-TIMER2) is used for fine power mosfet command implementation.
- Initialization procedures. These are needed for microcontroller clock and peripheral setup.
- Main scheduler. This is implemented in the main program, in order to be able to manage time deterministic function execution.
- Functions (CAN messages scheduler, heater MOS command controller, sensor monitoring).
- Control station. It allows the user to control the application software implemented within the smart skin. Moreover via control station it is possible to monitor all sensor data and diagnostic information.

Onboard a smart skin, the four heaters are labeled A to D. Each heater can be piloted with a law chosen among the following: threshold control, power target control, gradient control, and duty cycle control.

A note shall be made regarding gradient control: this name is misleading, since what this law actually does is targeting the time-derivative of temperature in a

single point, and not the space-gradient of temperature (i.e. difference between two points at the same instant). It shall also be taken into account that this control logic has been added as an experimental feature, and in fact it was not expected to perform exceedingly well, since it foresaw no temperature feedback. However, it is presented for completeness.

The remainder of this paragraph describes each control law more in detail.

A section of code, used to implement the threshold control for one heater (namely, heater A), is quoted hereafter (the same algorithm applies for the remaining three heaters).

```
void CommandThresholdTemperature ()
{
    if(u8FlagActivationThreshold[HEATER_A-1]==1)                (A)
    {
        if(AcqData.u16TempHeaterA>AcqData.u16UpperThHeaterA)    (B)
        {
            for(u8Cont=0;u8Cont<32;u8Cont++)
            {
                u8CmdTable[HEATER_A-1][u8Cont]=0;                (C)
            }

            DutyCycleAcquisition(HEATER_A,0);
        }

        if(AcqData.u16TempHeaterA<AcqData.u16LowerThHeaterA)    (D)
        {
            for(u8Cont=0;u8Cont<32;u8Cont++)
            {
                u8CmdTable[HEATER_A-1][u8Cont]=1;
            }

            DutyCycleAcquisition(HEATER_A,100);
        }
    }
}
```

Condition (A) verifies whether the threshold control for heater A is enabled or not (actually, each heater can be assigned a different control law). If the condition is true, the system checks if the measured temperature is greater than the maximum threshold (line labeled as B). If also this condition is true, then the duty cycle is set equal to 0% (code line C). On the other hand, if the measured temperature is lower than the minimum threshold (see line D), then the PWM duty cycle is set equal to 100%.

To further clarify the aforementioned code, it is useful to specify that, in order to set the PWM value, the system uses a 32x4 matrix (each row corresponds to one heater). The value is assigned by imposing the unity (1) value to a number of columns proportional to the desired duty cycle (e.g. DC=100% means a row containing 32 ones, while DC=50% means a row containing 16 ones followed by 16 zeros).

The relevant code segment used to implement the power target control for one heater, is quoted hereafter. As already said, the same algorithm applies for the remaining three heaters.

```

void CommandPower (void)
{
    int8    u8Step;

    if (u8FlagActivationPower[HEATER_A-1] == 1)                (A)
    {
        fCmdPowerIstantaneaHeaterA = (((float)AcqData.u16VSupply28 ...
            *0.029296875*(float)AcqData.u16VSupply28*0.029296875) ...
            *1000)/fHeaterResistance;                            (B)

        u16DutyCycleHeaterA = (int16)(((float)(u16InputPowerHeaterA))...
            /(fCmdPowerIstantaneaHeaterA)*100);                  (C)

        if(u16DutyCycleHeaterA > 100)
        {
            u16DutyCycleHeaterA = 100;
        }

        DutyCycleAcquisition(HEATER_A,u16DutyCycleHeaterA);

        u16OneHeaterA = u16DutyCycleHeaterA*32/100;              (D)

        for(u8Step=0; u8Step < u16OneHeaterA; u8Step++)
        {
            u8CmdTable[HEATER_A-1][u8Step]=1;                  (E)
        }

        for((u8Step = u16OneHeaterA); u8Step<32; u8Step++)
        {
            u8CmdTable[HEATER_A-1][u8Step]=0;
        }
    }
}

```

Similarly to what was done for the threshold command, the first line (A) is needed to check if the power target control is enabled for the specific heater. In case the condition is true, the maximum power rating for the heater is computed (B), and it depends on the smart skin feed voltage. Then, by dividing the desired power by the maximum available power and multiplying by 100, the equivalent duty cycle is computed (C). This is the PWM value that must be set in order to obtain the input (target) power value. The last action (line D) is to compute the number of ones to be put in the matrix row corresponding to the active heater, in order to obtain the correct duty cycle, and then the matrix is updated accordingly (code line labelled as E).

The following lines show the code used to implement the gradient control for one heater. Obviously the remaining three heaters use the same algorithm.

```

void GradienteCommand ()
{
    int16  u16Kostant = 3;
    int16  u16Vdd = 28;  // [V]

    int8  u8Contat;

    if (u8FlagActivationGradiente[HEATER_A-1] == 1)
    {
        u16DutyCycleHeaterA = ((int16)u8GradienteHeaterA/u16Kostant)...
            *(u16Vdd/(AcqData.u16VSupply28*0.0048828125));          (A)

        if(u16DutyCycleHeaterA > 100)
        {
            u16DutyCycleHeaterA = 100;
        }

        u16NumOfOneHeaterA = u16DutyCycleHeaterA*32/100;          (B)

        if (AcqData.u16TempHeaterA >= AcqData.u16UpperThHeaterA)    (C)
        {
            for(u8Contat=0; u8Contat < 32; u8Contat++)
            {
                u8CmdTable[HEATER_A-1][u8Contat] = 0;
            }

            DutyCycleAcquisition(HEATER_A,0);

            bOnOffCmdFlag = TRUE;
        }

        else if (AcqData.u16TempHeaterA <= AcqData.u16LowerThHeaterA) (D)
        {
            for(u8Contat=0; u8Contat < u16NumOfOneHeaterA; u8Contat++)
            {
                u8CmdTable[HEATER_A-1][u8Contat]=1;
            }

            for((u8Contat = u16NumOfOneHeaterA); u8Contat<32; u8Contat++)
            {
                u8CmdTable[HEATER_A-1][u8Contat]=0;
            }

            DutyCycleAcquisition(HEATER_A,u16DutyCycleHeaterA);
        }
    }
}

```

If the gradient target command is enabled, the algorithm computes a duty cycle value proportional to the required rate of change in temperature (improperly called

gradient). The goal is to ensure that, between a lower and an upper thresholds, the heater's temperature varies with a given rate. The equation accounts for the smart skin's feed voltage, so that a decrease in the measured voltage causes a balancing increase in the duty cycle (A). Then the system computes the number of ones to be put in the PWM matrix (B). Subsequently, the reference temperature is analyzed: if it is greater than the maximum threshold, then the heater must be OFF and the duty cycle is set equal to 0% (C); if it is lower than the minimum threshold, then the duty cycle is set equal to the previously computed value, in order to obtain a rate of change in temperature compatible with the required "gradient" (D).

Finally, a short extract of the relevant listings is reported hereafter, showing how the duty cycle control was implemented for one heater (of course, the other three heaters also use the same algorithm).

```
void CommandDutyCycle(int16 u16DutyCycle, int8 u8IDHeater)
{
    int8    u8Idx;
    int16    u16NumOne;

    u16NumOne=((u16DutyCycle) * 32)/100;                                (A)

    if(u8IDHeater == ALL_HEATER)
    {
        for(u8Idx = 0; u8Idx < 4; u8Idx++)
        {
            for(u8Count = 0; u8Count < u16NumOne;u8Count++)
            {
                u8CmdTable[u8Idx][u8Count]=1;                        (B)
            }
            for(u8Count = u16NumOne; u8Count < 32; u8Count++)
            {
                u8CmdTable[u8Idx][u8Count]=0;
            }
        }
    }
}
```

The algorithm is similar to the one presented for the power target control law, but in this case the duty cycle value is provided directly by the user and does not depend upon a given dissipation target. Therefore, very simply, at first the system computes the number of ones to be written in the relevant matrix row (A), and then it updates the PWM matrix accordingly.

2.4 ROV-E Smart Skin

Ideated as the heritage of the MULFUN project, ROV-E was conceived and proposed with an international consortium of six companies and research centers from all over Europe. The goal of the project is to develop multifunctional systems to be applied to exploration rovers. The European Commission selected ROV-E

as one of the studies to be funded in the 7th Framework Programme. Activities started in 2011 and will end in 2013, therefore the description contained in this section is not exhaustive.

Within ROV-E, Thales Alenia Space Italia aims at further improving the smart skin concept.

As already seen in the previous section regarding the STEPS Project, the smart skin is a new approach to electrical and electronic design of spacecrafts buses and exploration vehicles offering a fourfold function: routing of harness, hosting of distributed electronic, heating of surrounding structures and health monitoring. It is designed to be as lightweight, thin, flat and flexible as possible, in order to save space and mass and to adapt to different supporting frames. In particular, the ROV-E smart skin will:

- Host a monitoring network, sensing a number of different environmental parameters
- Host control electronics for payloads or instruments
- Host heating devices
- Route signal and (where possible) power lines
- Provide capabilities to communicate data with the “outside world” (main control unit and/or other smart skins)
- Provide a sufficient degree of electromagnetic compatibility
- Provide a flexible, compact layout for quick integration of all the aforementioned functions into an exploration vehicle or spacecraft

Ideally, the smart skin shall behave not merely like a flexible PCB, but like a complete device with specific functionalities. This means that the smart skin shall not simply be the backbone breadboard on which other rigid items are mounted, but instead it shall represent the flexible counterpart of equipments that up to now have been built in rigid form. In order to satisfy this requirement, the smart skin shall include not only conductor traces and vias, but also truly distributed electronics. As seen in the beginning of this chapter, today not all electronic components can be made flexible, nonetheless the industry is very near to this goal. In the meantime, the smart skin can mount some SMD items that could be replaced by bendable embedded versions in the future. The long term objective is the production of a single sheet with all circuitry embedded in it, from simple conductive traces to complex integrated circuits.

The new smart skin demonstrator will have in-plane dimensions in the order of 300 x 420 mm², and a thickness of a few tenth of mm. Up to now, there is no specific requirement on the overall shape, and therefore the baseline configuration is rectangular, but odd silhouettes with cutouts and peninsulas can be devised.

As usual, the first part of the research was dedicated to the definition of requirements. According to ROV-E requirements, the smart skin shall be thin, and its support layer shall not be thicker than 0.6 mm (TBC). It shall also be

flexible, capable of bending with a radius of 10 mm (TBC). These figures are easily attainable for the substrate in itself, but actual smart skin values shall be evaluated after integration of SMD components.

In the baseline design, the smart skin shall be made of polyimide. Alternative designs based on different materials can be proposed and investigated, provided that the new substrate undergoes a proper test campaign. The smart skin shall be based on conventional (silicon-metal) electronics. Optoelectronics can be studied at sample level as an option for further development.

The smart skin shall be built with the largest possible use of thick film technology and system-in-foil technology. All components that cannot be embedded with thick film technology, neither reduced to system-in-foil dimensions, shall be in the smallest SMD or THD packaging available and shall be surface mounted occupying the minimum possible footprint and space.

The smart skin shall route signal and power lines. In particular the flexible PCB shall embed all signal circuitry required for its operations, and all power lines required to feed the heating devices. The embedment of other power lines is still TBD, subject to maximum power transfer limitations in this architecture with flat conductor traces on flex polymeric substrate.

The smart skin control electronics shall communicate by means of a few-wires protocol. In particular, it shall be based on I2C/CAN bus. Moreover, it shall provide capabilities to communicate data with external subsystems (main control unit and/or other smart skins).

The smart skin shall host embedded heating devices, in order to be able to ensure minimum operative temperature for all its embedded electronics, and to control the thermal environment of surrounding structures. The smart skin shall also include a control logic for heater actuation in order to minimize power consumption.

The smart skin shall host a monitoring network, sensing a number of different environmental parameters. These parameters shall be available for logging, ground testing, and telemetry transmission.

The smart skin shall host control electronics for payloads or instruments, and the prototype shall include a representative working architecture and be able to pilot an external device at test-bench level.

The smart skin shall provide a flexible, compact layout for quick integration of all the aforementioned functions into the exploration vehicle or spacecraft. In particular, the smart skin design shall provide a monolithic piece to be bonded to a supporting structure and electrically connected by means of LIF/ZIF connectors.

Figure 2.38 shows a conceptual representation of the ROV-E smart skin.

The smart skin, when mounted on a support substrate in its final configuration, shall survive a mechanical environment typical of a launch vehicle, with particular attention to vibration loads.

The smart skin shall withstand bending during AIT without performance degradation, and it shall be able to operate in this bent configuration for its entire useful life (but it is not expected to undergo dynamic flexing during operations).

The smart skin shall provide a sufficient degree of electromagnetic compatibility. Its behavior in relation to conducted and radiated interferences shall be thoroughly examined. According to ECSS-E-ST-20-07C standard, in the 30 MHz to 18 GHz frequency range, electric-field radiated-emissions shall be limited according to Table

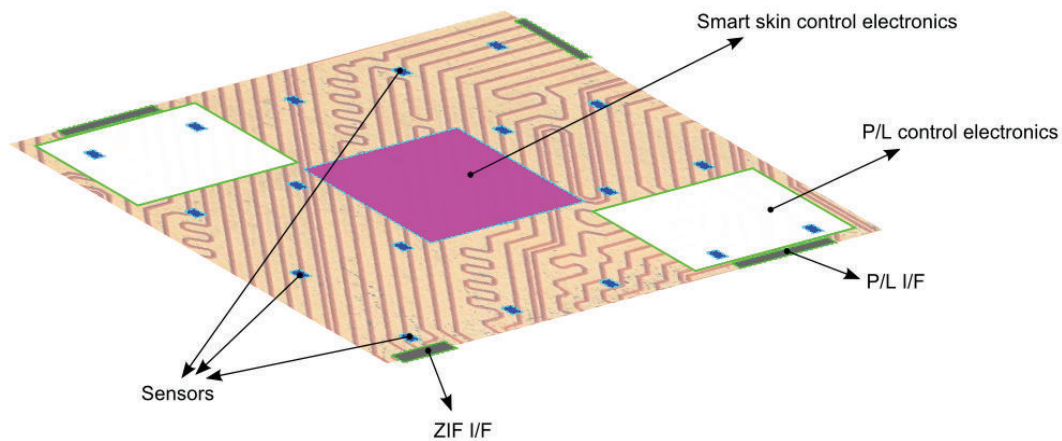


Figure 2.38: ROV-E smart skin concept, layout.

Table 2.24: EMC radiated emissions limits.

Frequency range	RE (dBuV/m)
30 MHz ÷ 100 MHz	60
100 MHz ÷ 18 GHz	+13 dB/decade

2.24 and Figure 2.39.

Notches due to launcher communication slots (e.g. telecommand - destruct reception or radar transponder transmission) are neglected.

The radiated susceptibility requirement is 2 V/m, i.e. the smart skin shall not show any damage, malfunction or deviation from the specified performance, when exposed to the 2 V/m susceptibility limit in the frequency range 30 MHz - 18 GHz. For test purposes, the injected sinewave signal shall be 50% amplitude modulated by 1 kHz square-wave. The requirement shall be met for both horizontally and vertically polarized waves.

During radiated susceptibility test campaign, in case of susceptibility, the test has to be repeated reducing the interference signal of 6 dB. The test report has to describe all performed tests and their results, in order to determine the actual susceptibility threshold.

Conducted differential mode current emissions in the frequency range 30 Hz - 100 MHz appearing on the smart skin power lines shall be within the limits of Table 2.25.

Table 2.25: EMC conducted differential mode emissions limits.

Frequency range	CE (differential mode)
30 Hz ÷ 10 kHz	100 dBuA
10 kHz ÷ 10 MHz	-20 dB/decade
10 ÷ 100 MHz	40 dBuA

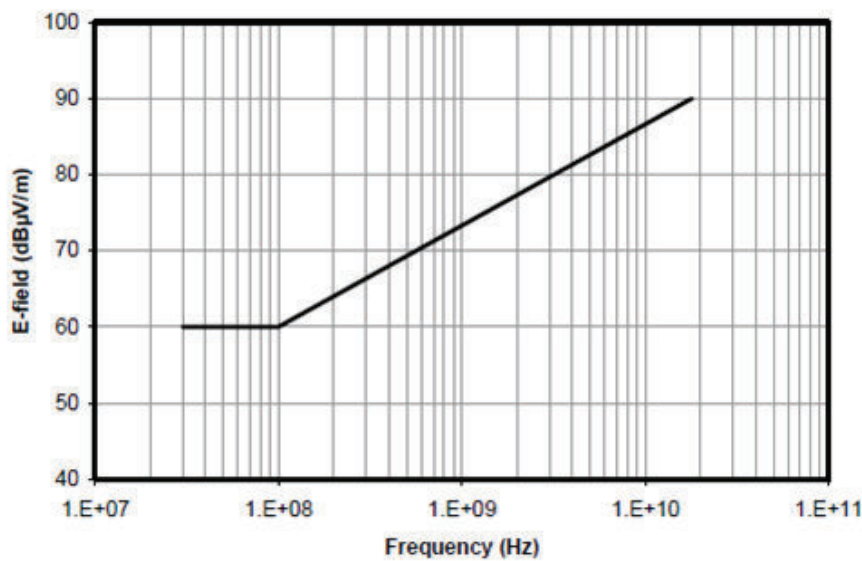


Figure 2.39: Smart skin radiated emissions requirements, based on ECSS-E-ST-20-07C, Figure A-3.

Table 2.26: EMC conducted common mode emissions limits.

Frequency range	CE (common mode)
10 kHz ÷ 100 kHz	80 dBuA
100 kHz ÷ 10 MHz	-20 dB/decade
10 ÷ 100 MHz	40 dBuA

Conducted common mode current emissions in the frequency range 10kHz - 100 MHz shall be within the limits of Table 2.26

Supposing a 28 V regulated power bus, the smart skin voltage emission between the individual power leads shall be less than:

- 560 mVPP for spikes (2% of nominal bus voltage)
- ± 70 mVP for ripple (0.25% of nominal bus voltage)

where the term spike designates a phenomenon with a duration lower than 10 μ s and no repetitive period, VP means peak voltage, and VPP means peak-to-peak voltage.

The smart skin shall not exhibit any failure, malfunction or unintended response when sine wave voltages of 1 V rms in the frequency range 50 Hz - 10 MHz are injected into the power bus inputs, or when the following common mode disturbing signal is superimposed on its input power line:

- Amplitude: $E=28$ V
- Duration: $\tau \geq 5 \mu$ s
- Rise time¹⁰: $t_r < 100$ ns

¹⁰Time needed for the injected voltage to change from 10% to 90% of nominal bus voltage.

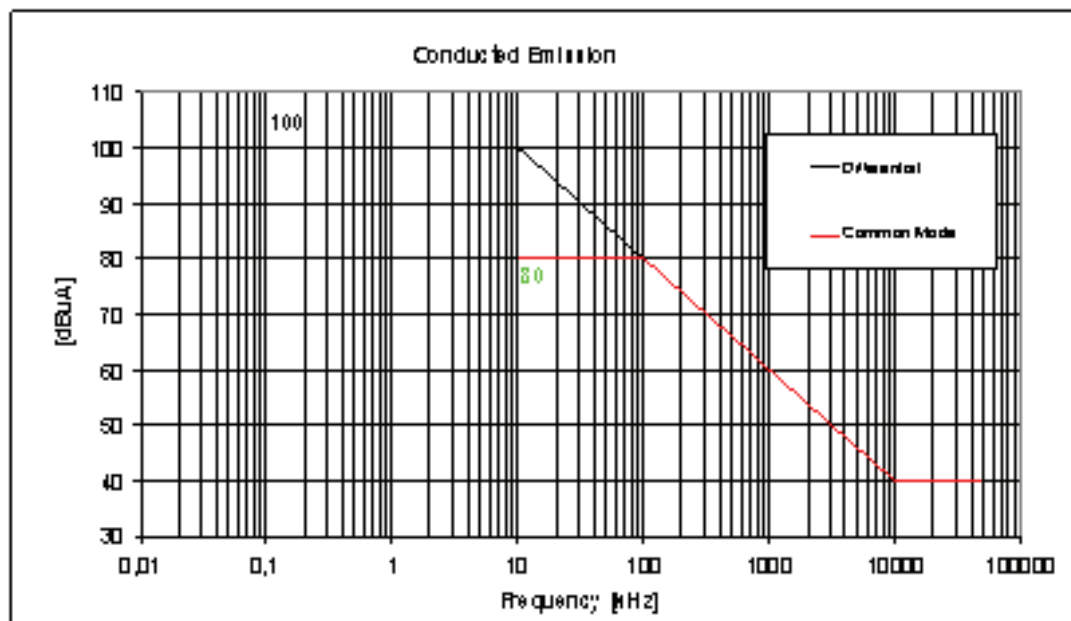


Figure 2.40: Smart skin conducted emissions requirements, common and differential modes, based on ECSS-E-ST-20-07C.

Refer to the graphic in Figure 2.41 for nomenclature clarification.

The smart skin shall not exhibit any failure, malfunction or unintended response when transient voltages with the following characteristics are superimposed on the primary power bus inputs.

- Transient voltage: $V_p = \pm 3V$
- Duration: $D = 15 \text{ ms}$
- Repetition frequency: 10 Hz
- $T < 10 \mu\text{sec}$ and $0.5 \text{ ms} < T < 1\text{ms}$

The injection shall be parallel between positive and negative lines, injecting both positive and negative going pulses (one at a time). Refer to Figure 2.42 for clarification.

During conducted susceptibility test campaign, in case of susceptibility, the test has to be repeated reducing the interference signal of 6 dB. The test report has to describe all performed tests and their results, in order to determine the actual susceptibility threshold.

The smart skin shall operate with nominal performance when exposed to conducted and radiated electrostatic discharge with the following characteristics (human ESD):

- Magnitude $> 10 \text{ kV}$
- Energy $> 10 \text{ mJ}$
- Peak current = 50 A

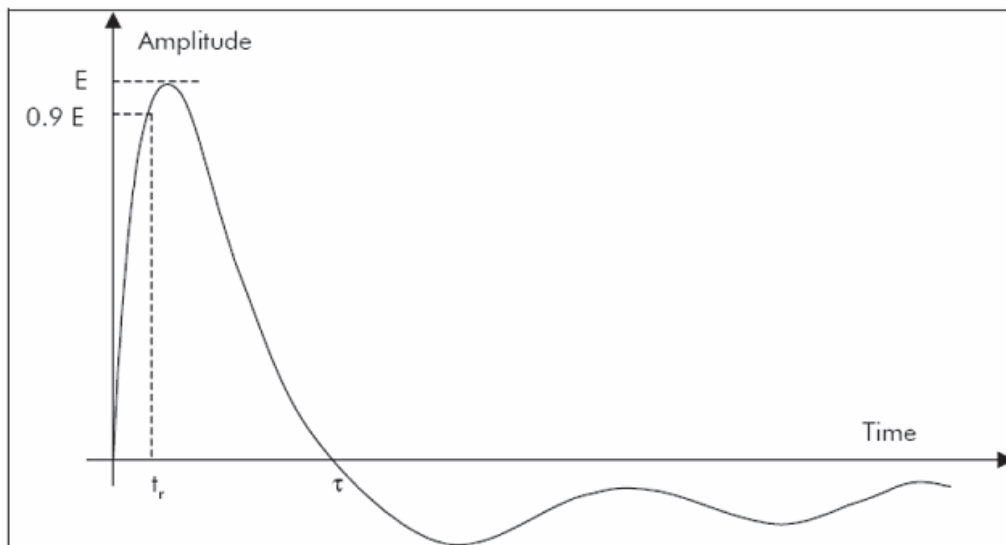


Figure 2.41: CM disturbing signal characteristics.

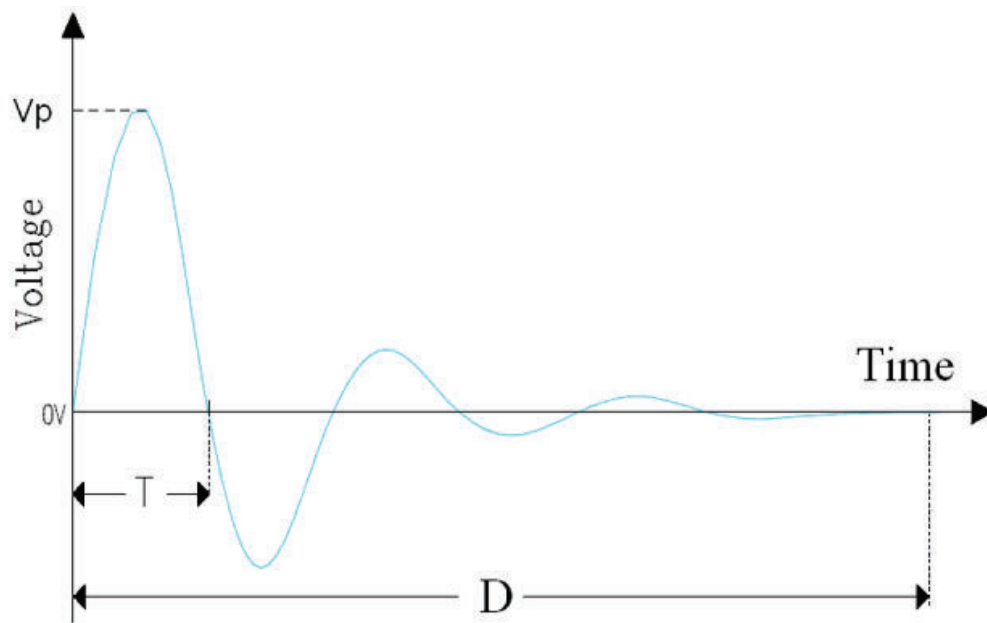


Figure 2.42: Transient voltges characteristics.

- Rise time (10% - 90%) < 10 nsec
- Duration (half amplitude) = 100 nsec
- Repetition rate = 10 Hz
- Duration > 3 min

Given the aforementioned requirements, the smart skin will be a flexible circuit, because usually flexible circuits reduce the size and weight of a finished product (thickness can be as low as 0.10 mm and weight reductions of over 75% have been achieved). Moreover, the flat planar nature of flexible circuits and the high track density achievable would allow higher circuit density and eliminate bulky connections and wiring, therefore offering considerable weight and space savings over traditional wire harnesses. Besides this, the ability to fold could expand the boundaries of design, installation (packaging) and operation. Indeed, unlike a rigid printed circuit board, which can only be positioned in two dimensions, flex circuits can be bent, twisted, and folded to utilize a third dimension. This property can be utilized in complex and tight-fitting assemblies where it would be impossible to accommodate several rigid boards with their harness.

Another positive remark is that production of flexible circuits is generally cheap, because it is based on low cost, high volume, reel-to-reel fabrication processes and techniques, and it relies on a variety of fabrication materials from low cost polyester to mid range PEN to high density adhesiveless polyimide. Moreover, circuits can be designed in configurations that span from simple, single-sided conductive paths to complex high density three-dimensional assemblies.

Primarily, there are four types of flexible circuits: single-sided, double-sided, multilayer, and rigid-flex.

Single-sided circuits consist of a single conductive layer and are the simplest, less expensive and most produced type of flexible circuit. Single-sided circuits are always specified for dynamic flexing applications such as printer and disk drive heads.

Double-sided circuits, as the name implies, are circuits with two conductive layers that are usually accessible from both sides.

Multilayer refers to circuits with more than two layers (see Figure 2.43). Interconnection between layers is usually ensured by means of plated-through holes. Double-sided and multilayer circuits cannot be used in dynamic flexing applications, since the copper would fatigue in a short time. Instead, double-sided and multilayer circuits are used where increased circuit density is needed and there is no dynamic flexing.

Rigid-flex circuits combine flexible circuitry with rigid board technology to create an integrated circuit that has almost endless applications. The rigidized area can be used to reinforce the circuit in an area subject to abrasion or other stress, or as an area where to bond analog or digital devices.

In general, the minimum bend radius capability of the final product is a function of its architecture: single and double-sided boards tend to have a minimum bend radius of ~ 6 times the thickness of the material, while multilayer flex circuits

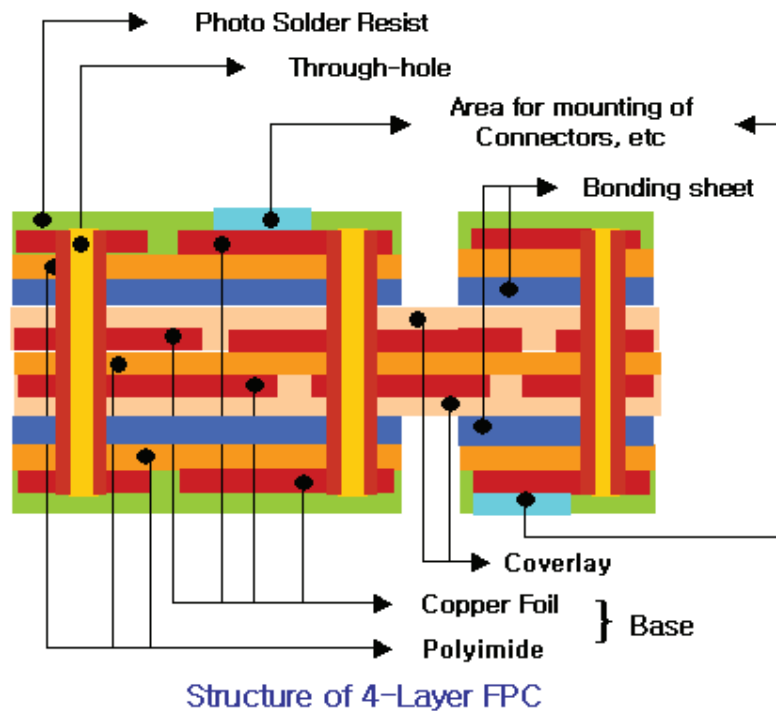


Figure 2.43: Example of multilayer (4 layers) flexible PCB (property of H.K. Best FPC Co., Ltd).

present a minimum bend radius of ~ 12 times the thickness of the substrate. These values are just guidelines and are referred to the bare PCB, at least 1.5 mm away from a plated through hole. Furthermore, additional PCB features (e.g. presence of stiffeners or of rigid ICs) can lead to an increase of these figures.

Given the fact that the smart skin should host a considerable number of components, it will probably need a multilayer design. A concept is depicted in Figure 2.44. At least in the specific ROV-E operative scenario (inner space of an exploration vehicle) it is plausible that the smart skin will not be subject to dynamic (continual or repeated) flexing, so there are no concerns on its fatigue resistance.

The relative position of conductive traces onboard the smart skin can help mitigating electromagnetic interference and can control impedance. For the moment, the assumption is to use a microstrip configuration with a single ground plane beneath uniformly spaced signal lines.

Moreover, to guarantee a minimal thickness and a reduced number of exposed (vulnerable) connections, the smart skin shall exploit to the maximum extent both thick film technology (or other printing technology able to create embedded flat passives) and system-in-foil technology. All components that will not be embedded into the smart skin substrate, shall be attached to the PCB (on one side only) with surface-mount technology mixed (if necessary) with through-hole technology. In particular, only components such as certain rare connectors and PGA ICs should need to be attached to the FPCB with through-hole technology. All other components should be mounted using surface-mount packages.

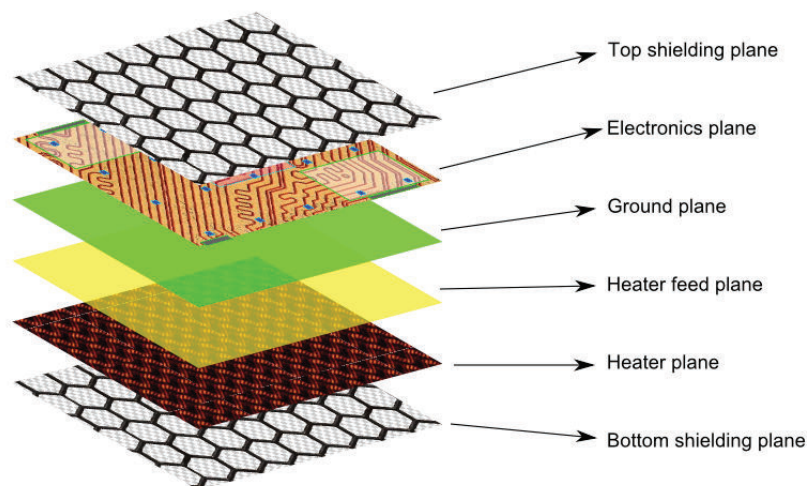


Figure 2.44: ROV-E smart skin concept, stack.

The surface mount package type (see Figure 2.45) is the mainstream choice for packaging electronic components of every type (including connectors). Its main characteristic is that all connections are made with a lap joint between a component lead and a pad on the surface of the PCB. This method has some clear advantages. For example, since the mounting procedure affects only the external surface and does not interfere with underlying layers, all inner (and reverse side) wiring space is available. Because of this, it is usually possible to wire a PCB in fewer layers than would be true with other connection methods. Another relevant benefit is the fact that surface-mount components are always smaller than their through-hole equivalent, making it possible to fit more parts in a given area.

The main disadvantages of surface-mount components stem from their small size. First of all it is more difficult to remove heat from SMT packages than it is for their through-hole equivalent (in some cases the heat generated is too high to permit proper operation in an SMD package). Secondly, one can not neglect the fact that it is hard to access SMD leads with instrumentation probes (diagnostic work and production testing become difficult).

On the other hand there is the through-hole technology (see Figure 2.46), which was the original package type used for electronic components. Through hole components are characterized by parts that have wire or formed leads which shall be inserted through holes drilled or punched in the PCB, and soldered to lands on the back side or to plating in the holes. A major benefit of through-hole components is the fact that every lead passes all the way through the PCB and therefore it has, intrinsically, access to any layer to make connections. Furthermore, every lead is available on the reverse side of the PCB, so test-probing them is easy. As already mentioned, with the advent of surface-mount components, through-hole is nowadays used primarily for connectors and for pluggable devices, such as microprocessors mounted in PGA packages.

Through-hole packages are often preferred for components that dissipate large amounts of heat, because of the relative ease with which heat-sinking devices can be fitted to them. In addition, it is much easier to provide a socket for a through-hole

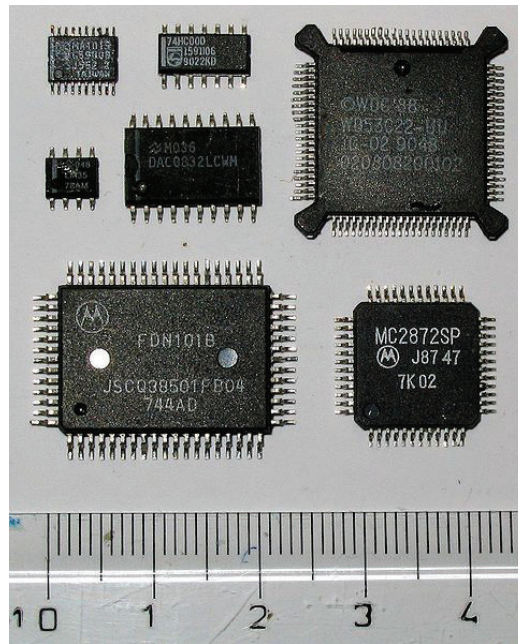


Figure 2.45: SMD samples (property of Wikipedia).

device and this eases the task of changing parts when it is necessary to upgrade a system. Nonetheless, integrated circuits in through-hole packages are becoming difficult to find as they are displaced by surface-mount equivalents.

Figure 2.47 presents a comparison of components built in SMD and THD package.

The mixed through-hole/surface-mount technique proposed for the smart skin is the most common method used to assemble electronics products. Assembly is a two-step process that involves placing all surface-mount parts and soldering them in place, then inserting all through-hole parts and soldering them in place (as an option, the through-holes may be hand-soldered if the quantity is very small).

All connections of the smart skin with external cabling, at least for signal traces, should be made via Zero Insertion Force (ZIF) connectors, to avoid problems and damages caused by applying force upon insertion and extraction. Low insertion force (LIF) sockets could reduce the issues of insertion and extraction but are not recommendable since the lower the insertion force, the less reliable the connection.

Unfortunately, ZIF sockets are much more expensive than standard IC sockets and also tend to take up a larger board area due to the space taken up by the hold/release mechanism. However, the need to find a handy and reliable connection method suitable both for lab research and for AIT activities can justify these drawbacks. Additionally, connectors shall have the smallest commercially available pitch (to reduce connector width for a given number of traces to be routed) and thickness.

Jumper or interconnect cables shall also be designed to reduce weight and overall space, thus suggesting to replace wire harness with Flat Flexible Cables (FFC) made of polyimide flexible strips. The FFC termination can be slightly thicker than the rest of the cable, in order to present a reinforced portion for ZIF interface.

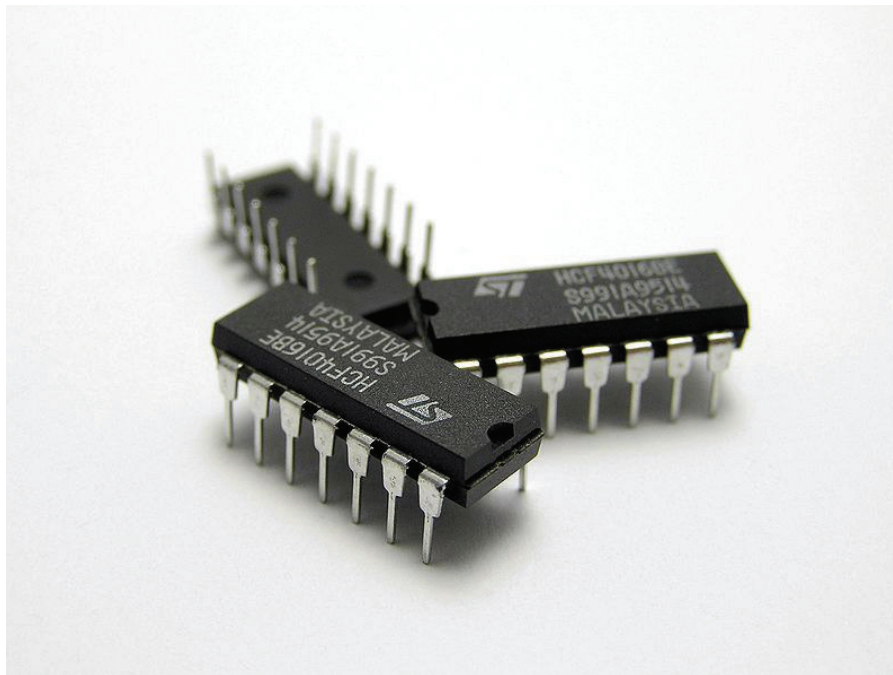


Figure 2.46: THD samples (property of Wikipedia).

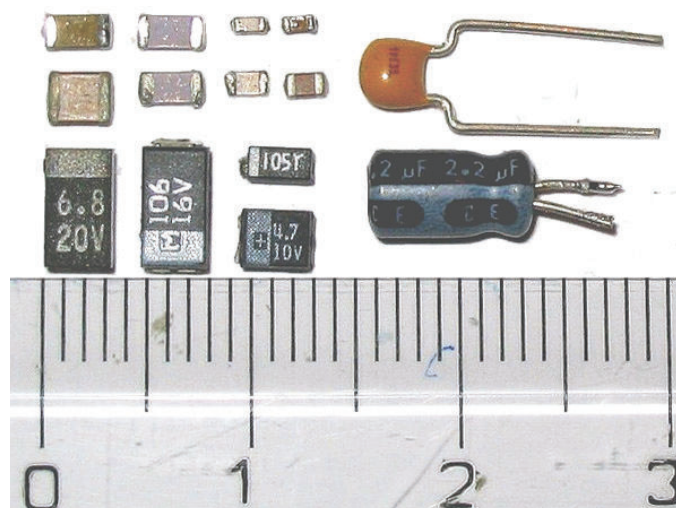


Figure 2.47: SMD - THD comparison made with capacitors (property of Wikipedia).

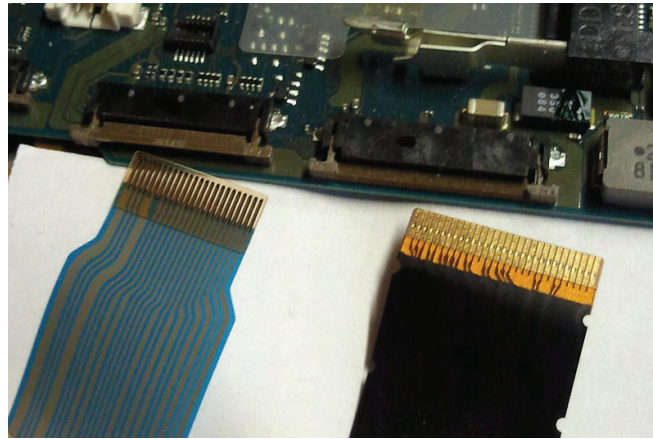


Figure 2.48: ZIF connectors and flat flexible cables (property of Wikipedia).

Figure 2.48 shows both ZIF connectors and mating FFC jumpers.

The PCB internal communication will be carried out in a few-wire protocol, very likely I2C or CAN.

Regarding the embedment of heaters in the smart skin, up to now two different technologies have been chosen to be analysed: Inconel heaters and printed heaters.

Inconel is a registered trademark that indicates a family of austenitic nickel-chromium superalloys. These materials are traded under different names: Chronin, Nickelvax and Nicrofer, to name just a few. However, they are often designated using the Inconel trademark followed by a number (e.g., Inconel 600: solid solution strengthened; Inconel 625: acid resistant, with good weldability; Inconel 690: low cobalt content for nuclear applications, and with low resistivity). Inconel alloys are oxidation and corrosion resistant, therefore they are well suited for service in extreme environments, and often they are used in high temperature applications. This is because, when heated, Inconel forms a thick, stable, passivating oxide layer protecting the surface from further attack; moreover, Inconel retains strength over a wide temperature range. As a drawback, Inconel is a difficult metal to shape and machine using traditional techniques. Welding Inconel alloys is difficult, too. This is due to cracking and microstructural segregation of alloying elements in the heat affected zone. However, nowadays several specially designed alloys can be welded via gas tungsten arc welding, electron beam welding, or pulsed micro laser welding.

At the time being, Inconel based foil heaters are very common. They are obtained from high temperature alloys, such as Inconel 600 or X750. Foil heaters are produced by bonding Inconel foils as thin as 0.001 in to a flexible insulating substrate such as silicone rubber or a polyimide film (i.e. Kapton). The resistive element is produced with a photo etching process that takes away the metal that is not part of the pattern. Then, the resistive layer is covered with another flexible insulating substrate foil. An example is shown in Figure 2.49.

At the present time, photolithography resolution is limited only by the wavelength of the light been used for exposure: ultraviolet excimer laser lithography allows minimum feature sizes down to 50 nm. The result is a thin, flexible, resistive patch that hosts a metallic resistor and can be bonded virtually to any surface, since their minimum bending radius is of the order of magnitude of 1 mm. These flexible

heaters have usage temperatures ranging from $-200\text{ }^{\circ}\text{C}$ up to $+150\div 200\text{ }^{\circ}\text{C}$, they can be applied in vacuum environments and are resistant to most chemicals (acids, solvents. . .). Their shape and size can be tailored to a great number of different design requirements. Their power density rating depends upon the heater's insulation, presence of internal adhesive, heat sink control temperature, and mounting method. As a rule of thumb, the watt density at room temperature ranges from 0.4 to 7.5 W/cm^2 (depending upon insulating material and presence/composition of adhesive between foils). In general, higher power densities are achieved with solutions having the drawback of a certain amount of mass loss and outgas in vacuum. They are often thinner than 0.3 mm and weigh around 0.05 g/cm^2 .

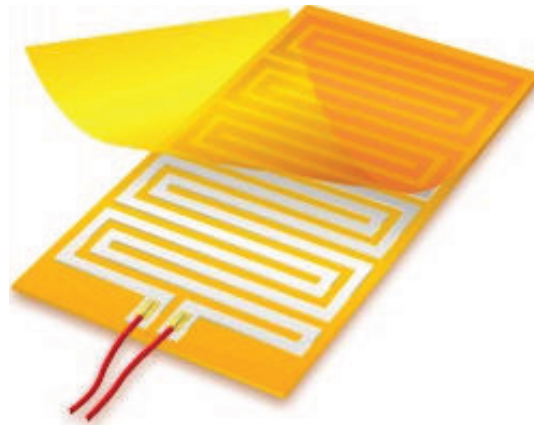


Figure 2.49: Etched foil heater (source: MINCO website).

On the other hand, printed heaters can be built with thick film polymer technology. Layer thickness ranges from 5 to several hundred microns (typical values are about $25\text{ }\mu\text{m}$: thick film heaters offer a 2D-profile, and can be fitted into restricted spaces. Due to low thermal mass and excellent heat transfer, thick film heaters are a well established solution in a number of industrial and commercial applications, where they have proven to be energy-efficient heat sources, leading to space and energy savings. Another advantage of thick film heaters deals with quality issues. In fact, due to the heating element structure, traditional failures due to moisture adsorption and electrical leakage are avoided.

Printed resistors can be manufactured with accuracies up to $\pm 5\%$ without supplemental trimming process. Multiple screen-printing passes enable an expansion of the ranges of the printed resistors, e.g. three repetitions of screen-printing with different carbon-based inks provide three orders of magnitude range, from 100 ohm to 100 Mohm .

Polymer thick film heaters have a temperature range for continuous operation going from -60°C to 200°C . In fact, maximum temperature of operation depends upon the substrate and the dielectric thickness, and certain solutions can go up to 600°C , but this poses strong limitations on materials selection.

Moreover, different resistor paste formulations enable a wide range of resistance and coefficient of thermal expansion, providing several options in terms of circuit design and product behaviour (ref. Table 2.27).

Generally speaking, resistors are built trying to ensure that their resistance remains as stable as possible independently from the temperature. On the other

Table 2.27: Different properties for different paste formulations.

	Pd/Ag	Ni/Cr	Mo	C
Specific resistance (Ohm/sq)	0.1 ÷ 0.2	0.2 ÷ 2.0	50 ÷ 1000	2 ÷ 200
CTE (ppm/K)	300 ÷ 3500	400 ÷ 600	-400 ÷ 400	-500 ÷ -1000

hand, it is possible to manufacture resistors that have not only a resistance which is linearly variable with temperature, but also a well defined temperature where the slope of the line changes abruptly (resistance increases or decreases suddenly). Such a resistor, whose resistance varies significantly with temperature, is called thermistor, a portmanteau of thermal and resistor. Materials with a relatively rapid increase with temperature (i.e. a higher coefficient) are the most interesting under an engineering point of view. The higher the coefficient, the greater the variation in electrical resistance for a given temperature increase.

The correlation of resistance change with temperature change can be either negative or positive. If the resistance decreases with increasing temperature, the device is called a negative temperature coefficient (NTC) thermistor. If the resistance increases with increasing temperature, the device is called a positive temperature coefficient (PTC) thermistor.

NTC thermistors are used as resistance thermometers (especially in low-temperature measurements of the order of 10 K) and as inrush-current limiting devices in power supply circuits (they prevent large currents from flowing at turn-on, and then allow higher current flow during normal operation). PTC thermistors can be used as current-limiting devices for circuit protection (as replacements for fuses) or as simple, reliable and inexpensive timers (e.g. in the degaussing coil circuit of most CRT displays and televisions).

But, for the sake of the discussion, the most interesting application of PTC resistors is another one. In a PTC thermistor, as the temperature of the conductive medium increases, so does the electrical resistance. If the external voltage is kept fixed, the result is a diminishing current, and at the same time a diminishing power output for each temperature increment. In other words, PTC resistors adjust their power output in response to the surrounding temperature. In simpler terms, they are self regulating heaters (see Figure 2.50).

A self temperature regulating and self heat output regulating heater has some advantages. First of all, it has the tendency to reach a limiting temperature and this increases safety, because it ensures that the temperature of the object being heated constantly stays within an ideal range (steady thermal conditions are reached by modulating the power at any change of the heat transfer conditions), and it operates without the risks of overheating or burnout (the PTC mechanism stops the current absorption when any abnormal poor heat transfer condition occurs). It has high electro-thermal conversion rate and ensures even heat distribution and low energy

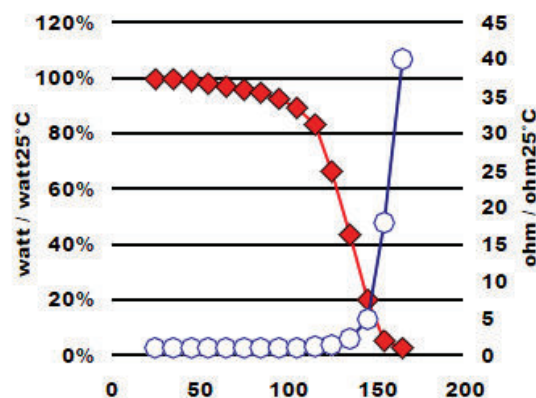


Figure 2.50: PTC behaviour: horizontal axis shows Celsius temperature (Zoppas Industries R&D bulletin).

consumption.

In particular, PTF PTC heating elements present several attractive features, due to their:

- Thin section: a thin cross-section profile allows compact sizes and free flow surfaces in fluidic applications.
- Light weight: a low thermal mass provides instantaneous thermal response.
- Mechanical flexibility: direct application on surfaces to be heated maximises heat transfer.
- Inherently distributed nature: power spread on large surfaces guarantees safer low watt densities and temperatures.
- Self-regulating properties: self-regulating temperature behaviour increases element safety in abnormal conditions, potentially eliminates thermal protectors (and related RAMS issues, AIT issues, costs...), optimise power consumption to heating request, is not sensitive to voltage variations.
- Design adaptability: applicability onto many different substrates can ensure a simple heater integration on many different PCBs or equipments, while flexibility in circuit configuration provides control of the applied power distribution, and grants reliable and efficient redundant designs.

Table 2.28 summarizes typical properties of PTF PTC heaters.

In any case, with both Inconel or printed heating devices, it is mandatory to obtain a good and complete embedment into the smart skin. The baseline idea is to print them with resistive serigraphic pastes in a dedicated layer. The possibility to use PTC materials is still under discussion, given the fact that they could offer the advantage to self-regulate the heater output according to the local heat transfer conditions, optimizing power dissipation.

From a geometrical point of view, TAS-I is currently studying a particular layout that could enable high modularity, high reconfigurability and precise targeting of areas to be thermally controlled. As a general precaution, parallel circuit designs

Table 2.28: Typical properties of PTF PTC resistors.

Resistance range (ohm/sq)	1000-9000	
TCR range (ppm/°C)	5000-300000	
Switch off temperature range (°C)	50 - 150	
Heat up rate (°C/sec)	up to 100	
Useful power density (W/cm ²)	in still air:	<1
	in forced air:	<2
	for water heating:	<4
	In vacuum:	n.a.
Maximum operative temperature (°C)	Polypropylene, Polyethylene:	100
	Polyester (Mylar):	130
	Polyimide (Kapton):	180
	Nylon, PTFE, PPS:	180
	Epoxy/Glass, Mica:	180
	Glass, Ceramics, Metals:	180

will be preferred, since they can preserve the heating function even in case of a local damage of the flexible printed circuit board.

Regarding the distribution of functions on different layers, the initial proposal is represented in Figure 2.51. This is an ideal situation in which each layer performs just one single function. However, increasing the layer count decreases flexibility and adds manufacturing problems. Therefore, the actual smart skin stack will be determined taking into account the supplier's constraints and will probably be made of just three or four copper layers.

The external smart skin layer shall be an insulating, protective coverlay and, perhaps, a shielding device (i.e. metallic fabric). A solid or patterned (cross-hatched) shielding plane made of copper or screen-printed conductive ink could be a good baseline. An interesting alternative is a conductive coating, like silver epoxy (it is more flexible than copper, but perhaps raises contamination issues).

Several techniques have been evaluated for EMI/EMC mitigation.

Electrical and magnetic fields, as well as sudden electrical discharges and ionizing radiations, can hamper the functionality of electronic equipment. In general, a wide spectrum of phenomena, globally indicated as Electromagnetic Environmental Effects (E3), should be taken into account in order to design a reliable and robust electronic item. E3 cover EMC, RFI, EMI, EMP, ESD, RADHAZ, lightning, and the like.

In the specific smart skin case, the focus is primarily on EMI and EMC aspects.

Electro Magnetic Interference (EMI) occurs when an electrical disturbance from a natural phenomena (e.g., lightning or ESD) or an electrical/electronic system

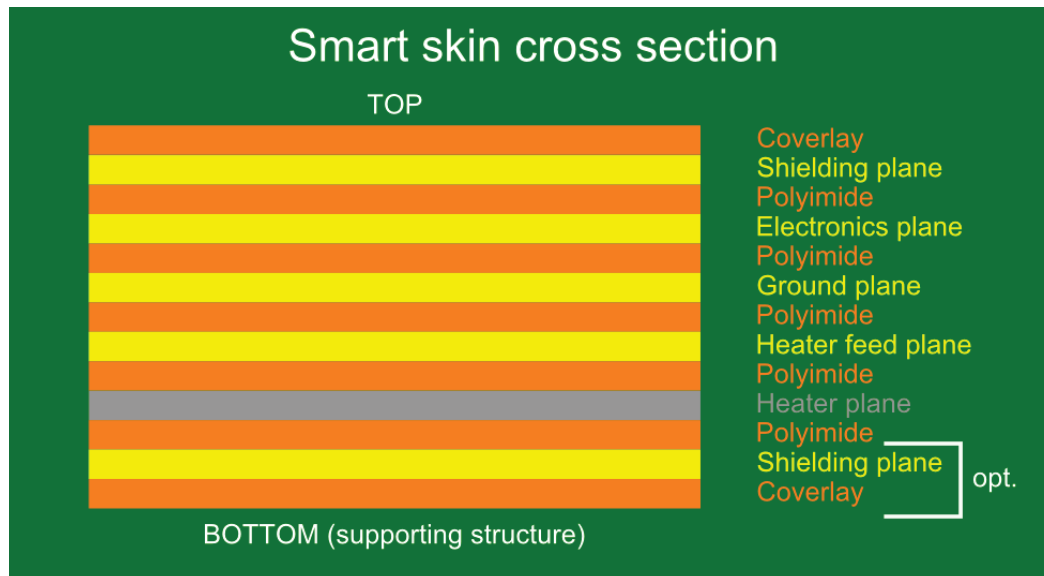


Figure 2.51: Concept for the smart skin cross section stack.

causes an undesired conducted or radiated response in a victim device. EMI refers to any conducted, radiated or magnetically induced voltage which degrades, obstructs, or repeatedly interrupts the desired performance of electronic equipment.

Electro Magnetic Compatibility (EMC) is the status in which operations of equipments and systems in their installed environments cause no unfavorable response to or by other equipments and systems in the same environment. In order to keep under control the electromagnetic behavior of electronic devices, there are four quantities to be monitored:

- Conducted Emission (CE), which is the potential EMI generated inside some equipment and carried through the I/O lines, the control leads, or power lines.
- Conducted Susceptibility (CS), which refers to the EMI coming from the outside of a piece of equipment to the inside over the I/O cables, signal leads, or power lines.
- Radiated Emission (RE), describing the potential EMI which radiates from escape-coupling paths such as cables, leaky apertures, or inadequately shielded housings.
- Radiated Susceptibility (RS), referring to undesirable EMI radiated into equipment from external electromagnetic sources.

EMI emission, the unintentional or undesired exiting of potentially interfering electromagnetic energy from electrical/electronic sources, always occurs. In fact, all electronic devices give off electromagnetic radiation when they are operating and this can affect other electronic devices: this unwanted emission of electromagnetic waves from a device and its reception of unwanted electromagnetic radiation can interfere with normal operations.

The amount of electromagnetic interference (EMI) given off by a device can be minimized through good circuit design, adequate grounding, and the use of

suitable filters. However, this may not be sufficient and a shielding may be added for further protection. The main objective of any type of shielding is to prevent the passage of electromagnetic waves into or out of the device. EMI shielding can work by reflection, absorption, or by carrying the electromagnetic radiation to ground. Its performance is measured by shielding effectiveness, which is the relative ability of a shield to screen out undesirable electric and magnetic fields and plane waves. Shielding effectiveness is measured as the ratio of the signal received without the shield to the signal received inside the shield. It is often expressed in decibels.

The fundamental principles exploited for EM shielding are to establish a Faraday cage or to absorb energy.

Metals were the original EMI shielding materials and metal casings were, for many years, the only method of shielding electronic devices. Metal cans placed on the circuit board are still used. However, rigid metal casing is not a viable option for the smart skin design, since it would nullify any benefit coming from mechanical flexibility and mass reduction.

Fortunately, there are many other methods that can be explored to provide a shielding function, namely:

- Shielding substrate compounds (wire oriented silicone, conductive elastomers. . .)
- Conductive foils (mainly metallic sheets and tapes)
- Conductive fabrics (metallized fabric, neoprene or silicone filled woven metal screen. . .)
- Conductive lacquers (conductive painting, electroless plating, vacuum metalizing. . .)

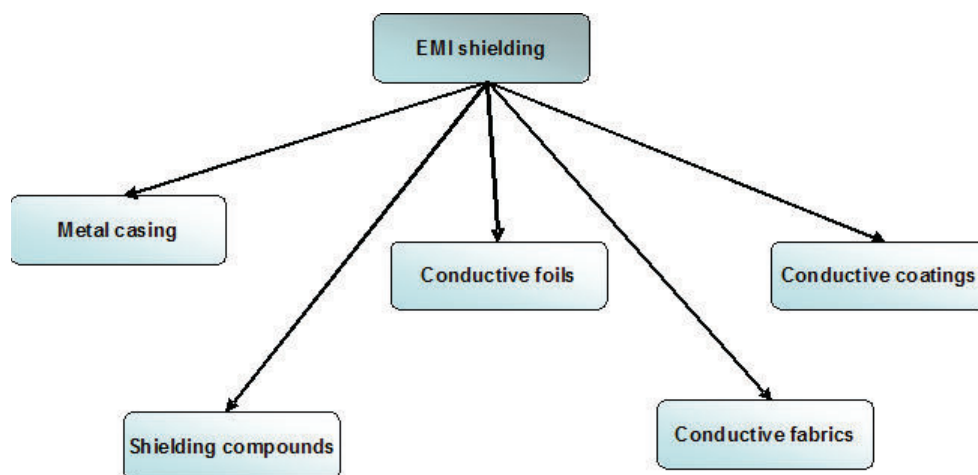


Figure 2.52: Shielding techniques for the smart skin.

Conductive compounds are mainly doped or wire oriented polymers.

Conductive plastics shield by absorption of the radiation in the bulk of the material. They can be obtained from almost any plastic material, but thermoplastics and elastomers (silicone or fluorosilicone) are preferred. They work by including either conductive metals or conductive metallized fibers into the bulk of the material.

The fibers can be either metal or metal coated (stainless steel, copper, and carbon - alone or nickel coated), they have a high aspect ratio and form a continuous conducting network inside the bulk plastic to provide the EMI shielding. In this way, the conductivity is in the structure of the board itself, rather than simply on the surface. In this case, there is no relationship between the surface resistivity and the shielding. The shielding effectiveness depends on the volume resistivity of the plastic. The volume resistivity is usually between 10^{-2} through 10^1 ohm-cm. Due to this volume effect, the actual degree of shielding effectiveness provided by the finished part is influenced by the thickness of the part, by the overall conductivity level of the modifier (conductive additive) used in the compound, and its dispersion. In addition to shielding, these compounds can provide some form of ESD control if required by the application.

Recently, carbon nanofiber/LCP composites have also been investigated as promising electromagnetic interference shielding materials.

Shielding compounds can be processed in injection molding or extrusion applications although the method of adding the conductive fibers is different. Conductive plastics are available die-cut, molded or as sheets.

Conductive plastics have significant advantages over conductive coatings. First of all, the EMI shielding is an integral part of the board and cannot be scratched or delaminated. A second point is that shielding is already present with the product itself, without the need for additional treatments (this implies very low set-up costs and lead times, making them cheaper and sometimes faster for small to medium production volumes). The overall plastic compound can also include flame retardants or other additives to enhance some properties of the final product which are not strictly related to EMI/EMC aspects. Conductive plastics (in particular, conductive elastomers) are made to satisfy requirements of maximum shielding in a severe environment: these typically pertain to all military and aerospace applications. However, in other industrial sectors, the arguments are not entirely positive for conductive compounds, because for very high volumes (i.e. greater than 1000000 parts), the cost of setting up a coating processes can be amortized sufficiently to make them very competitive with conductive compounds. Moreover, a conductive compound should be passivated before conductive traces deposition takes place (otherwise it could short electrical paths of the circuit). This is the same step required for thin, flexible metallic substrates (that would provide a shielding thanks to Faraday effect) so, from this point of view, there is not clear advantage of conductive plastic substrates on metals.

Another particular class of shielding devices is that of wire oriented plastics (silicones). They are an intermediate breed between shielding compounds and shielding fabrics. They use the Faraday cage principle. These materials are produced from the combination of specification grade plastics (mainly, silicone) and wire. The silicone elastomer acts as an environmental seal, while the wires provide excellent conductivity to establish EMI/RFI integrity. They are particularly useful in gasket applications requiring high levels of attenuation along with a moisture seal. However, given the fact that they are rather economic, they can also be used in less demanding commercial shielding devices. They are produced with individual aluminum or monel (nickel - up to 67% - and copper) wires in both solid and sponge silicone. The version with the silicone sponge elastomer is designed for

applications which have severe joint unevenness, require low closure forces, have a ~ 5 psi maximum operating pressure and require a greater compressibility than solid silicone. On the other hand, the version with the silicone solid elastomer is designed for applications requiring moderate closure forces, high operating pressures and requiring a wider temperature range. These materials are regularly used where long service life, high performance fixed gaskets are needed. Because of the harder nature of the material, and because it needs a strong closing force through a relatively rigid cover (possibly with screw fixed at regular intervals), it is not generally suitable for doorseals or other applications where a regular opening and closing is required.

Wire oriented plastics can be shaped in sheet and strip forms, and some manufacturers also offer custom compositions and shapes on request. Their allowable temperature range is determined by the elastomeric substrate, in general solid silicone can withstand -60°C to 260°C , while sponge silicone can be used in the range -75°C to 205°C .

Shielding (conductive) foils, sheets, and tapes are as simple as the name suggests: thin sheets of metallic foil used for both shielding and grounding.

Commercially available metal foils are made of aluminium or copper. Thickness is minimal, on average ~ 1.5 mil. Due to the dead-soft nature of the foil, it can be used almost everywhere, even on irregular and curved surfaces. To enhance this capability to line nonconductive boxes, cabinets, and walls, foil sheets are usually adhesive backed.

Similarly, foil tapes are coated with an extremely aggressive acrylic adhesive and can be used for sealing boxes, joints and ducts (against dust and moisture) and for masking. They offer a cost-effective solution to EMC shielding, even for single ICs. The tapes can also be used for ground termination between panels on shielded rooms. Other interesting applications are repair and modification of printed wiring boards, flat conductor assemblies, EMI/RFI shielding of small electromagnetic components (transformer and reactor coils, instrument and control motors), prototype wiring board design and layout, and shielding of cable slices or cable repair.

The base metal is subject mainly to maximum temperature limitations depending on its melting point, while versions with adhesives provide a much narrower service range (more or less -15 to 50°C).

Conductive fabrics are shielding fabrics made of “woven” wires. These wires are metal threads or yarn, or conductive-coated yarn: usually, they are made of highly flexible nickel over copper or copper plated polyester.

Typically, the wire diameters range from 0.025 mm to 0.5 mm and the weave densities vary from 50 to 200 openings per inch (OPI) mesh. The mesh can be set at any angle up to 45° . Absorption of electromagnetic energy is mainly dependant on the mesh conductivity, wire diameter, weave density, electrical contact between intersecting wires, mesh material and edge terminations. As a rule of thumb, shielding effectiveness is around 75-80 dB @ 1 GHz. These materials have good weatherability: their operative temperature range is -40°C to 70°C , they can survive cold exposure without cracking and can operate in wet/salty conditions.

Sometimes, shielding fabrics can be filled with neoprene or silicon to give a solution similar to wire oriented plastics (e.g. Alumashield).

Conductive lacquers work by establishing a Faraday cage on the surface of the product to be shielded. Regardless of the type of conductive medium, there are some common issues involved: good adhesion to the substrate (may require a pre-treatment like etching, or the selection of substrates with low production stresses), correct area coverage (may require masking), durability and weatherability of the coating (scratching, rubbing or thermal cycling can create cracks and damages with loss of continuity of the shielding effect and possible birth of slot antennas), considerations about the recycling, reuse, or effective disposal of the manufacture. Applied layers are often in the range $0.02 \div 0.04$ mm for a shielding effectiveness >75 dB.

Conductive paint was one of the first methods used for EMI shielding of plastics and is still one of the most popular methods of providing EMI shielding. They are made by dispersing conductive fillers (silver, silvered copper, copper, nickel) in a variety of paints to provide a continuous conductive network.

Copper and silver based paints provide good EMI shielding against electrical fields, but are not as effective against magnetic fields. Nickel based paints are resistant to scratching and have a lower level of conductivity than other paints. Graphite-doped paints also provide a good EMI shielding, but the final layer tends to be weaker with respect to metal formulations (however, polishing and heating, besides reducing the surface resistivity, also make the weak graphite layer more firm and adhering). Given the fact that the binding agent is volatile, when graphite coatings are used in vacuum, the lacquer must be heated on forehand.

Acrylic based paints have often better adhesion than water based ones. Almost all shielding lacquers are not compatible with contaminants like water, oil, and greasy flux residues. These paints are sprayed onto dry, clean surfaces in layers up to 80 micron thick (but $1 \div 2$ mils can be sufficient). They are usually dry to touch in a few minutes and cure at low (room) temperature in a day. As with any spraying process, “over-spray” is inevitable and this is generally in the region of 30-40%. This is paint that is lost to the process and can increase the cost of the process for complex products where multiple spray passes are needed for the required conductive paint thickness. New spraying techniques such as high-volume, low-pressure (HVLP) spraying can reduce “over-spray” considerably. Some of these paints can even be used to shield entire rooms.

Electroless plating is a method of attaching a thin coating of metal to a clean, dry substrate to provide the shielding. Unlike electrolytic plating, electroless plating does not need an externally applied current for the process and operates autocatalytically from solution. Electroless plating can be used as either a double-sided coating where the plating is over the complete product or as a single-sided coating (selective coating) where the coating is applied only to selected areas of the board (in this latter case, the product is pre-treated with a base coat and then masked to ensure that coating only takes place where desired). Usually the coating is composed by a layer of copper (up to 3 microns) which provides the conductivity and main EMI shielding plus a top layer (up to 1 micron) of electroless nickel alloy which provides corrosion resistance and wear protection for the copper layer. Electroless plating provides good EMI shielding against electrical fields. This can be further improved by increasing the thickness of the copper layer, either by continued electroless plating or by conventional electroplating. However the process is not very effective

against magnetic fields.

Vacuum metallization applies a thin coating of aluminium (layer thickness up to 10 microns) to the dry, clean surface of the product, in order to provide a highly conductive layer and a very effective EMI shielding. Aluminium is used because it is inexpensive, it has a very high conductivity, and it is highly corrosion resistant in most environments. Vacuum metallizing can be used for complex shapes, but often requires a mask to ensure that the metal is deposited in the correct locations. Vacuum metallizing can also be carried out on thermoformed parts to form the equivalent of a metal can for board level EMI shielding. However, this latter solution is not applicable to the smart skin, since it has the same disadvantages of rigid metal enclosures.

Silver epoxy is a single component epoxy based substance containing very minute silver particles suspended throughout the medium. It can be screen printed, brushed or sprayed. Some relevant mechanical and electrical properties are listed in Table 2.29. This material allows for electrically conductive paths to be placed on non-conductive surfaces and therefore provides an alternative method for shielding from electromagnetic interference.

Silver doped epoxy has some advantages, e.g. it enables a reduction in layer count and thus reduces thickness (layer thickness is approx. 0005 in and this in turn increases flexibility when compared to using copper as the shield material). Silver epoxy also reduces cost due to the fact that this material needs less processing.

However this material has also some disadvantages: i.e., when it is “screen applied” the accuracy of locating the image is not as precise as that offered by the methods used to produce printed copper shields (generally ± 0.015 in is required for feature tolerance), and silver epoxy cannot be used as an electrical connection inside a plated through hole.

As a final comment, it is interesting to give a look at Table 2.30, making a comparison between conductive coatings and shielding compounds.

Besides the use of proper shielding materials, good design choices can help mitigate EMI/EMC problems. Ground and power plane distribution can effectively shield the circuitry from radiated interference, equally spaced peripheral bonding points, connecting a ground layer to the spacecraft common ground, can also be helpful. Other interesting techniques have been proposed for space-grade cables equivalent to a Twisted Shielded Pair (TSP) [31]. A first solution is to use a cross pattern on two layers connected by a large number of vias. Unfortunately, these vias can introduce losses and degrade the signal at high frequencies. Another approach is the introduction of side shielding. The basic idea is depicted in Figure 2.53.

Single traces or pairs of conductors that need shielding are shown as embedded in the polymeric substrate. These traces are shielded on the bottom by a continuous ground plane and on top by smaller ground plane areas. A laser is used to cut grooves through the polyimide from the top down to the bottom ground plane, and these grooves are then copper plated or filled with a conductive ink. In this manner, a signalling pair has full circumferential shielding and the two conductors minimize the loop area when transmitting a signal. The conductors, here shown in side-to-side configuration, can also be arranged in a stacked fashion, to make the loop area even smaller.

Table 2.29: Silver epoxy properties.

Property (units)	Value
Sheets resistivity (ohms/sq/mil)	0.012
Bulk resistivity (ohm-cm)	$3 \cdot 10^{-5}$
Adhesion strength (psi)	
Tensile	1500
Lap shear	2000
Thermal conductivity (cal/cm/sec/°C)	0.01
Specific heat (cal/g/°C)	0.07
Coefficient of thermal expansion ($^{\circ}\text{C}^{-1}$)	$3 \cdot 10^{-5}$
Modulus (tensile, psi)	$6 \cdot 10^6$
Poisson's ratio	0.35

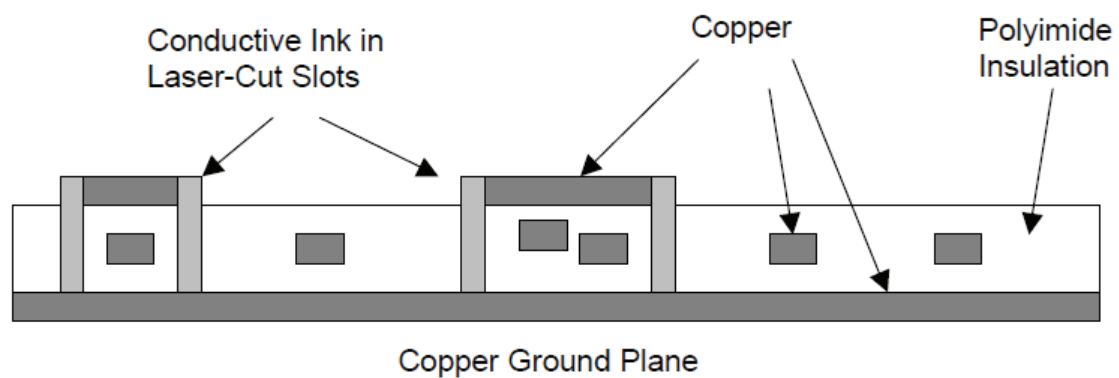


Figure 2.53: Cross-section view of metallic side shielding added to flex circuit cabling [31].

Table 2.30: Comparison between shielding coatings and shielding compounds (tradeoff made by RTP company).

	EMI shielding compounds	Conductive coatings
Shielding performance	Integral property	Scratching the coating may create slot antennas (shield failure) May delaminate during thermal cycling and have other adhesion problems
Recyclability	Reusable and recyclable	Stripping process removes coating, but this creates metallic sludge which must be discarded
Part design compatibility	Complex designs	Simpler designs (LOS process)
Lead times	Lower than plastics with conductive coatings	Higher than EMI shielding compounds
Corrosion resistance	Integral property	Copper, favored by highly conductive coatings, requires protective topcoat
Post-mold shielding operations	Integral property: part shielding right out of the mold	Additional shielding steps include: coating part with conductive material and masking
Special handling during molding	None	Must be kept free of contaminants for coating to adhere properly

With respect to the smart skin electrical design, some reliability aspects are already under consideration. A generic electronic assembly is composed of a printed circuit board, soldered joints, and electronic components of several types and packaging. In generic electronic assemblies failures could occur in each of the above mentioned parts. Possible failures can come from delaminations (any planar separation within a printed board, like a separation between plies, either within a base material, or between base material and a conductive foil), blistering (any delamination in the form of a localized swelling, with separation between any of the layers of a the base material, or between base material and conductive foil or protective coating), and poor solderability (solderability is the property of a metallic surface to be wetted by molten solder). Causes of delamination and blistering are incorrect or partial inner-layer joining and/or curing processes, presence of foreign material at the interfaces of PCB pre-preg material, excessive moisture absorption by the printed circuit material. Possible causes of bad solderability are solder bath contamination, poor thermal conditions, poor solder quality and mainly poor PCB solder pads finish. Regarding delamination and blistering, since the first two causes are only controllable by the FPCB manufacturer, the only countermeasure that the end-user may apply is the one indicated by ECSS, that is to say, demisting the printed board in a static oven (temperatures between $90^{\circ}\text{C} \div 120^{\circ}\text{C}$, for minimum 4 hours) or in a vacuum oven for temperatures lower than 90°C , and storing the PCB in dry nitrogen, dry air, vacuum or desiccants (storage in ambient conditions, i.e. clean room, is allowed for less than 8 hours). To avoid solderability failures, the design will pose some specifications regarding the nature of PCBs pads, in particular recommending the use of fused electroplated Sn/Pb surface finishes over copper. Viable alternatives can be ENPIG (Electroless Nickel/Palladium and Immersion Gold) or ASIG (Autocatalytic Silver and Immersion Gold).

To enhance the mechanical resistance during operations, the flex circuitry must be stress relieved, and, if let loose in certain points (as has been done, for example, with the ABB Dallas sensors cables), it must be detuned to avoid resonance during vibration testing or launch. These goals can be achieved by using drops or lines of glue on the flex circuit runs: usually a simple RTV silicone adhesive is well suited for this job. Additionally, if the flexible motherboard or some appendages have to be routed underneath a bulky chassis, there must be protection against vibrational chafing.

The latest version of the block diagram of the smart skin is presented in Figure 2.54

2.5 JPL MFS-based μ Rover

As seen in the section dedicated to Stablcor, recent developments in printed circuit board fabrication processes have led to products with increased strength and stiffness, by incorporating single or multiple graphite weave layers within the PCB stack during the lamination fabrication process. A typical printed circuit board is composed of prepreg and copper foils (laminated to prepreg on both sides). When using a Carbon Core Laminate, this is incorporated into the PCB construction as a pre-laminated layer (carbon fiber weave and binding resin in the middle, sandwiched

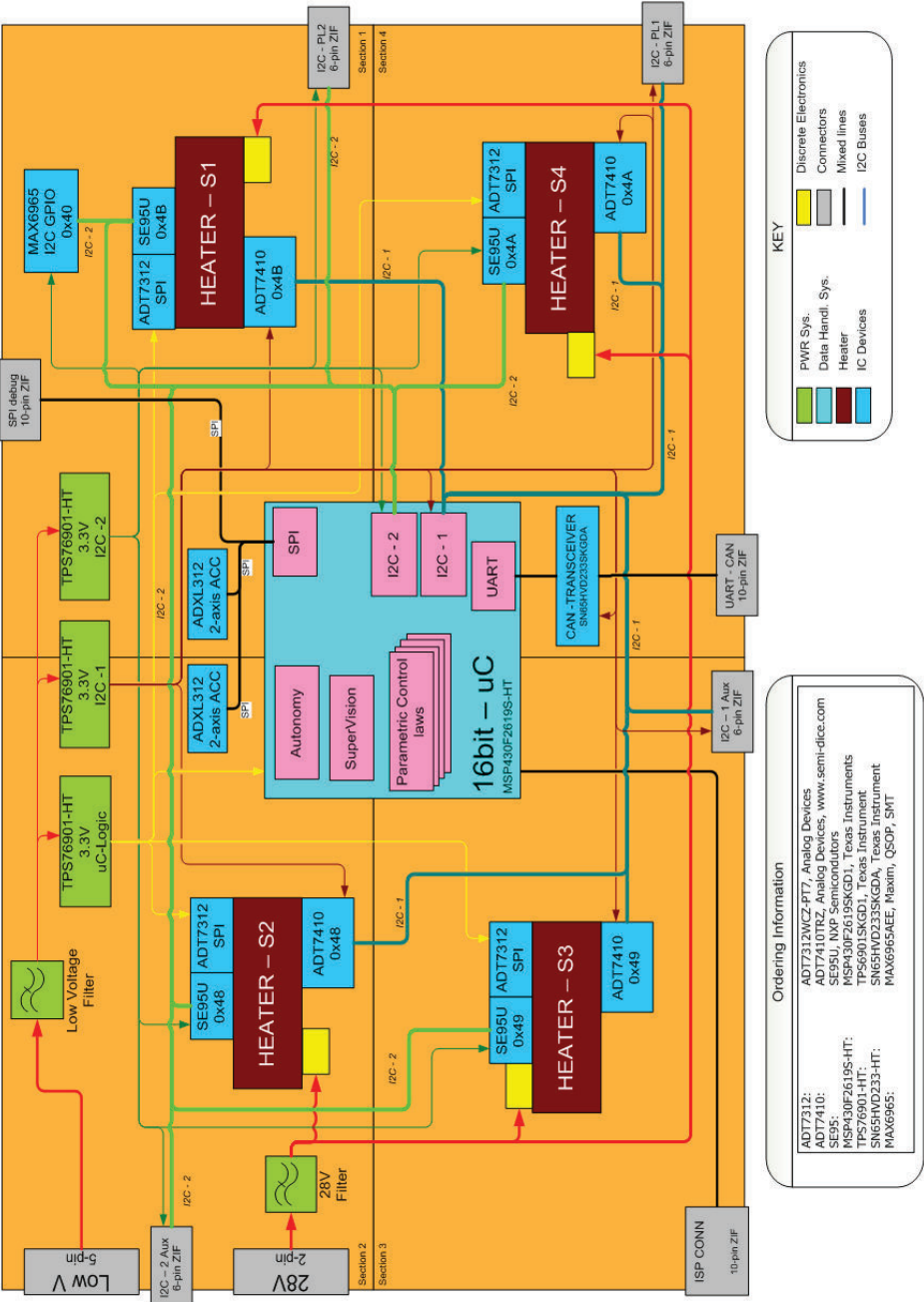


Figure 2.54: Block diagram for the ROV-E smart skin.

between copper foils on both sides).

This new technology brings together a mechanical and a thermal benefit. It could enable a reductions in mass through the elimination of metal spars, metal enclosures, panels, backplanes, and cabling. Moreover, it could provide the ability to implement high density local data processing capability directly at or near instruments such as cameras or sensors.

It was already remarked that one of the main limitations of conventional rigid printed circuit boards is their being intrinsically bidimensional. The goal of the activity carried out with the Jet Propulsion Laboratory was to conceive a working tabletop demonstrator of a a multi-functional system characterized by the exploitation of planar geometries to build more complex (tridimensional) elements.

Past work on mechanical performances [65] conducted at JPL (already quoted in the previous chapter and in the section dedicated to Stablcor) shows that printed circuit boards, when designed properly, can have substantial mechanical strength and stiffness, with some additional desirable properties like tailored CTE and good thermal conductivity. Of course, connecting these different planar geometries into a 3D product requires ad-hoc solutions, like for example the use of structural adhesives and fasteners to achieve effective structural members that can compete with metallic structures.

The main objective of the research activity was therefore to design and build a micro rover with multifunctional characteristics, thanks to the use of these state-of-the art PCB materials. As additional requirements, candidate functions and technologies to be included were:

- Basic mobility (forward motion, backward motion, and turn capability)
- Distributed electronics (circuits spread on the walls of rover's body, where PCBs act as structural panels)
- Ability to build the most part of the rover out of a 2D geometry
- Presence of a gripper (which may or may not be mounted as end-effector on a manipulator arm)
- Presence of a laser pointer to simulate a scientific P/L
- Presence of contact sensors
- Usage of compliant mechanisms instead of conventional ones
- Wireless communication with control station
- Inclusion of parts fabricated with a stereolithography process

A preliminary activity was a thorough review of small existing rovers: both space- and non space-oriented devices were considered. A total of 127 products were investigated, gathering information on size, mass, and locomotion system. The scope of this research was to collect data to steer the conceptual design phase. A small exploration rover designed at JPL for deployment on asteroids, MUSES-CN, was selected as the best concept example. The MUSES-CN rover was an experiment

of rover mobility and miniaturization, where a huge integration effort was required from the beginning of the design. Therefore MUSES-CN had to face all the drivers that can justify the use of multifunctional systems. Moreover its simple shape and reduced dimensions made it the ideal comparison for the JPL MFS μ Rover. More details on MUSES-CN can be found in the appendix.

After selecting a reference vehicle, setting the basic requirements, and creating a conceptual design, the first substantial development activity was the electrical, mechanical, and software design of the small rover system.

All the structural parts making up the rover, except four angular brackets, are conceived to be multifunctional PCBs. They can be manufactured in flat configuration, on the same raw panel, and then routed out to allow integration of the rover. The body enclosure is the most cumbersome item, and it is made of a single rigid-flex printed circuit board, that can be folded to create a box structure (see Figures 2.55 and 2.56). The envelope of the rover's body is a parallelepiped measuring max 150 x 150 x 80 mm³. The rigid islands of this body are equipped with circuitry on both sides and can accommodate all the components needed to make the rover work. In the current design, only SMD components are considered, but the technology is perfectly capable to host quad flat packs, micro ball grid arrays, and flip chips among others. In an actual space flight applications, electronic components can be located inside the box, protected from a potential harsh environment. If the environment is not an issue, locating components both inside and outside the box could maximize processing power or even help with thermal management. Cabling normally associated with electronic interconnection is eliminated by means of the flexible parts of the rigid-flex body.

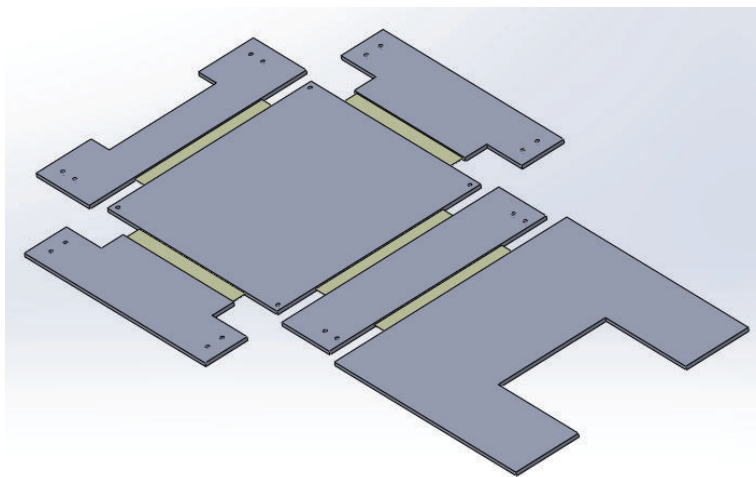


Figure 2.55: Rover body in unfolded 2D configuration.

Besides the main body, other parts of the rover that can be obtained from the same 2D PCB are links for the manipulator arm and wheels (at least part of them). All these parts can have circuits, components and connectors mounted on them, exactly like the main box.

In order to form the structural body of the rover, two problems had to be solved: i) to keep the rigid-flex board in the desired 3D configuration, and ii) to accommodate the flex interconnections avoiding damage. To accomplish these two functions, the

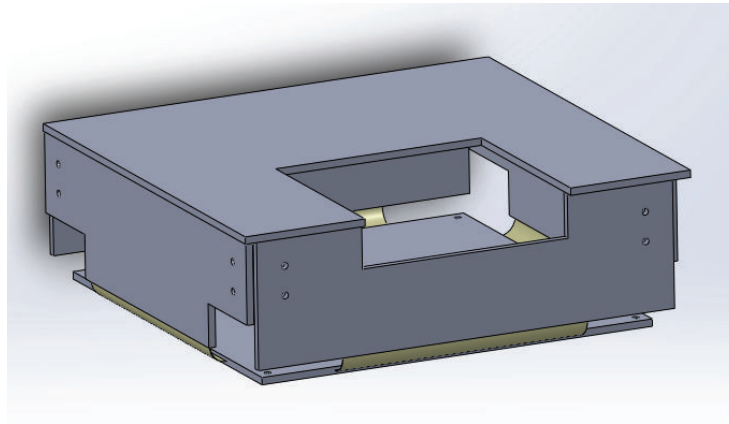


Figure 2.56: Rover body in folded 3D configuration.

body can be either encased (glued or fastened) in a cubic frame or assembled with prismatic spacers. This application is targeted to a small demonstrator, and, as a secondary goal, wants to explore the applicability of stereolithography to rapid prototyping and manufacturing. Therefore the spacers and brackets needed to put together the final assembly have been built in ABS plastic thanks to a Dimension SST 1200ES 3D printer¹¹. ABSplus plastic has an elastic modulus of 2 GPa, a tensile strength of 36 MPa (5200 psi), a tensile elongation of 4%, and a flexural strength of 52 MPa (7604 psi).

Besides the brackets or frame needed to keep the main body together, several other components can be fabricated with a stereolithographic process:

- Gripper
- Compliant mechanisms
- Journal bearings
- Treads for wheels
- Rover body (PCB) mock-up

Indeed, a whole intermediate mock-up was built with the aforementioned 3D printer: the components proved to be of good quality, able to reproduce design features down to 0.762 mm, and they were an invaluable tool in design troubleshooting and software development.

To ensure a safe folding of the structure, the minimum bending radius r_{min} for flex interconnects must be carefully chosen. There are design rules, in FPCB manufacturing world, that allow to compute the minimum bending radius from an empirical factor (6 to 25, depending on the number of layers or application) multiplied by h , the overall height of the flexible part:

¹¹This printer can take as input a 3D printing STL file that can be obtained from any parasolid model. The build envelope is 254 x 254 x 305 mm, and the machine uses Soluble Support Technology (SST). i.e. the support material is removed at the end of the fabrication process thanks to an alkalide bath.

- For 1 layer PCBs: $r_{min} = 6 \times h$
- For 2 layers PCBs : $r_{min} = 10 \times h$
- For multilayer PCBs: $r_{min} = 15 \times h$
- For PCBs subject to dynamic flexing: $r_{min} = 25\div 40 \times h$ (increase in bend radius normally increases life)

Taking into account the worst case of a 400 μm thick 4-layers interconnect, a 6 mm bending radius (0.4 mm x 15) is assumed as safe.

Using the rover's main structure to host the electronics frees up the inner volume, where other components can be accommodated, in particular: a battery, servo motors, and a WiFi antenna.

Two possible mobility architectures were evaluated: a first one with motors hosted inside the rover and wheels moved through shafts (supported by means of plain bearings - pivot bushings), and a second one with motors encased inside the wheels. Both architectures are suitable for individual-wheel drive layout (4 servos required), while the one with motors hosted inside the rover also allows for a reduced number of motors (2 servos required, shafts on same side connected by belts, chains, or gears). It is worth mentioning that the solution with motors in wheels could free up more space inside the small rover box. However, the first architecture was chosen, deciding to exploit the servomotors themselves, with their casing and internal bearings as wheel supports, therefore simplifying the design. The rover is equipped with four wheels. All of them are powered and they are directly mounted to the rover's body (no legs and suspensions). Differential steering is used: thanks to fixed position and orientation of the wheels, this is a mechanically simple solution, and at the same time it ensures a maneuverable vehicle able to turn on the spot. The main disadvantage of differential steering is the high friction resistance due to slippage of wheels that translates into a need for higher power to turn the vehicle (however, being very lightweight, the rover will likely suffer a reduced impact of this phenomenon).

The rover is also equipped with a five joints robotic arm, completed by a compliant gripper and a laser mounted as end effectors. For the arm assembly, the main concern is the dimensional envelope, both in terms of size and weight. It should be small enough to be retracted on top of the rover box. The structure (including motors) should also be light and short enough to be actuated with existing hobby servos, characterized by low torque values, and it should give the motors a significative mass margin to lift a P/L.

One possible solution for a small and light arm is to include a reduced number of degrees of freedom, to reduce length, weight (less joints = less motors), and complexity. However, the choice was to keep an high number of DOFs and work on the proper selection of actuators and materials to reduce weights; the arm can therefore move with:

- Waist yaw (around a vertical axis)
- Shoulder pitch (around an horizontal axis)

Table 2.31: Summary of servo motors used in JPL μ Rover. Listed torques are stall values measured at 6 V. Current values are averages (maximums are ~ 1200 mA).

Joint	Servo	Motion range	Torque	Current	Weight
Wheels	Parallax Continuous Rotation	$0 \div 50$ rpm	2.8 kg cm	150 mA	44 g
Waist	Parallax Standard	$0 \div 180^\circ$	2.8 kg cm	150 mA	44 g
Shoulder and elbow	Hitec HSR-5498SG	$0 \div 180^\circ$	10.6 kg cm	200 mA	60 g
Wrist (pitch and roll)	Hitec HS-85BB “Mighty Micro”	$0 \div 180^\circ$	3.0 kg cm	200 mA	20 g

- Elbow pitch (around an horizontal axis parallel to the shoulder joint)
- Wrist pitch (around an horizontal axis parallel to the shoulder and elbow)
- Wrist roll (around an horizontal axis perpendicular to all the previous)

All the servomotors are 4 to 6 V rated and controlled in Pulse Width Modulation (PWM).

As mentioned before, at the end of the robotic arm there is a compliant gripper. The study and application of compliant mechanisms is one of the key research areas at JPL, and therefore this μ Rover had the goal of demonstrating integration of a compliant mechanism at system level. Three possible applications were evaluated for the inclusion of this technology in the demonstrator: gripper actuation, opening and closing of compartments, or bi-stable actuation of probes. The first option was chosen.

The geometric design was initially inspired to a design proposed in literature [82], and then substantially modified to allow the integration as a end effector for the rover. The geometry of the part is presented in Figure 2.57. Its dimensional envelope is $\sim 90 \times 68 \times 19$ mm. The compliant links are 8 mm high and 0.8 mm thick. The gripper has been modeled in Solidworks as a monolithic component made of ABS plastic, with a modulus of 2 GPa, and a yield strength of 36 MPa.

The gripper has an open-close tweezer movement, normally open. It is actuated by means of three linear stroke Shape Memory Alloy (SMA) motors, Miga NanoMuscle NM706-Super, connected to the central “bridge” part so that they can pull it back toward the rover body. The NanoMuscle motors employ wires made of Nitinol (Nichel Titanium Naval Ordinance Laboratory) shape memory alloy as force generators. These materials undergo a solid-state phase change when cooled below a specific transition temperature, then return to their original geometry when heated, providing constant power output across the entire stroke length. The specific design of these motors, called “folded construction” (see Figure 2.59), mechanically

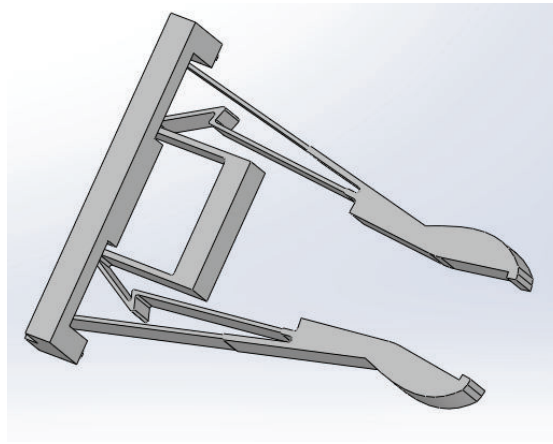


Figure 2.57: View of the layout of the compliant gripper.

amplifies the displacement, without any corresponding decrease in force. It consists of a series of SMA wires attached to a stack of rigid plates. The plates are coated with a low-friction material so that they can slide easily against one another as the wires heat up and contract. Given that the plates are mechanically connected in series, the stroke length of the actuator is $(n-1)$ times the contraction of a single wire, where n is the number of plates. The return force must be provided by a spring, in this case replaced by the tendency of the elastic gripper to go back to its relaxed undeformed configuration. SMAs operate on a cycle of heating, cooling, and deformation: NanoMuscles have cycle times ranging from 320 to 1600 msec, and they are rated to more than 1 million cycles. These motors are small (see Figure 2.58), lightweight, and silent: they are a good substitute for rotary actuators and related mechanisms.

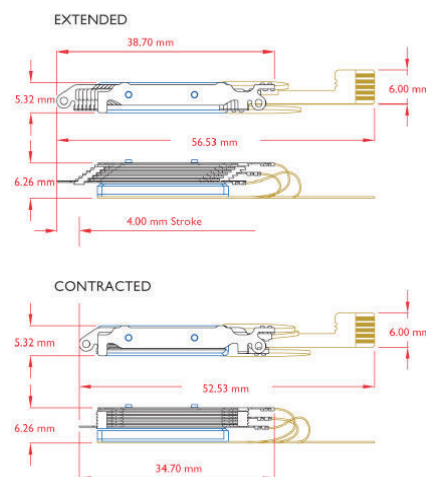


Figure 2.58: Dimensional characteristics of Miga NanoMuscle motors (property of MIGA Motors).

As mentioned before, the three shape memory alloy actuators apply a load on the central “bridge” part of the gripper. The behavior of the CM has been studied both with a nominal load of $1.3N$, equal to the full rated pull of two SMA actuators, and with a maximum load of $2.1N$, equal to the maximum BOL pull of three SMA

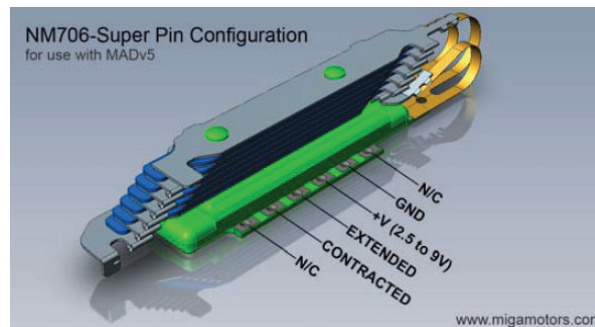


Figure 2.59: Layout of Miga NanoMuscle NM706-Super motors (property of MIGA Motors).

actuators. Moreover, the analysis has been made both in a free configuration, with nothing obstructing the movement of the claws, and with extremely stiff spring connectors interposed between the claws, in order to simulate the presence of an incompressible payload being held. These boundary conditions are represented in Figure 2.60.

The mechanical behavior of the gripper was studied with the Solidworks Simulation analysis tool, tailoring the solver settings to the particular case under examination. The compliant structure required a finer mesh because it must undergo large, nonlinear deflections. Moreover the finite element analysis must be displacement-controlled, and therefore the large displacement option was selected.

The results for the $1.3N$ load case, with no spring connectors between claws, are depicted in Figures 2.61, 2.62, 2.63, and 2.64. They represent the stress status of the gripper (Von Mises and X direction, which is the longitudinal symmetry axis), the displacement caused by the load, and the strain field.

A second case has considered a $1.3N$ load, and the presence of an object between the claws. Results are shown in Figures from 2.65 to 2.68.

A third case has considered a $2.1N$ load, and the presence of an object between the claws. Results are shown in Figures from 2.69 to 2.72.

Then a different case has been designed to study the out-of-plane deflection of the gripper when bearing a payload. The presence of an object being lifted was simulated by means of a distributed 0.3 N force on the terminal portion of the claws, as depicted in Figure 2.73. Results are shown in Figures from 2.74 to 2.76.

To complete the overview, the buckling behavior of the gripper has been evaluated, both under the nominal $1.3N$ load (with and without presence of spring connectors), and under an asymmetric point load. The first three buckling modes have been extracted, and the corresponding Buckling Load Factors (BLF) checked. The lowest actuation load causing buckling would be $\sim 4N$, while the smallest point load causing buckling would be $0.75N$.

After the characterization of the compliant gripper, to complete the description of the end effector, it must be told that a $5V$, $5mW$, 650 nm laser module is mounted on the last link of the manipulator arm, beneath the gripper itself.

The design of the whole μRover was modelled in Solidworks®. All COTS components were sketched, and the structural members needed to compose the rover were designed. With all these parts available, more than one digital mock-up

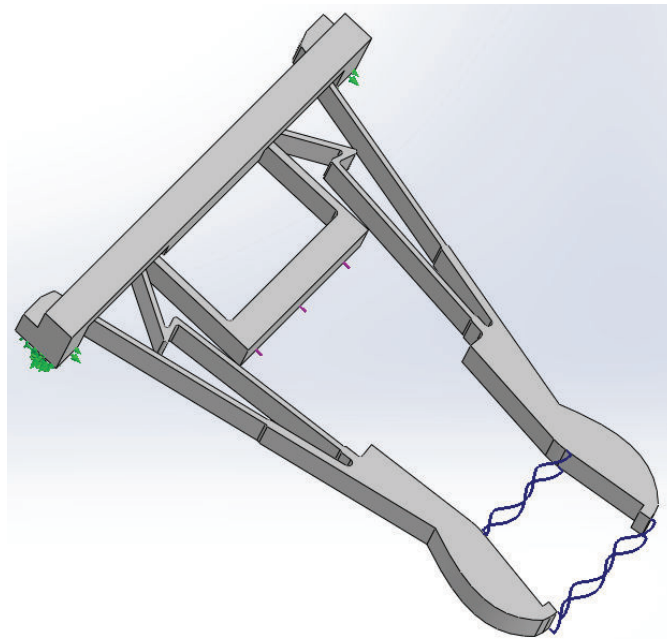


Figure 2.60: Boundary conditions on the gripper.

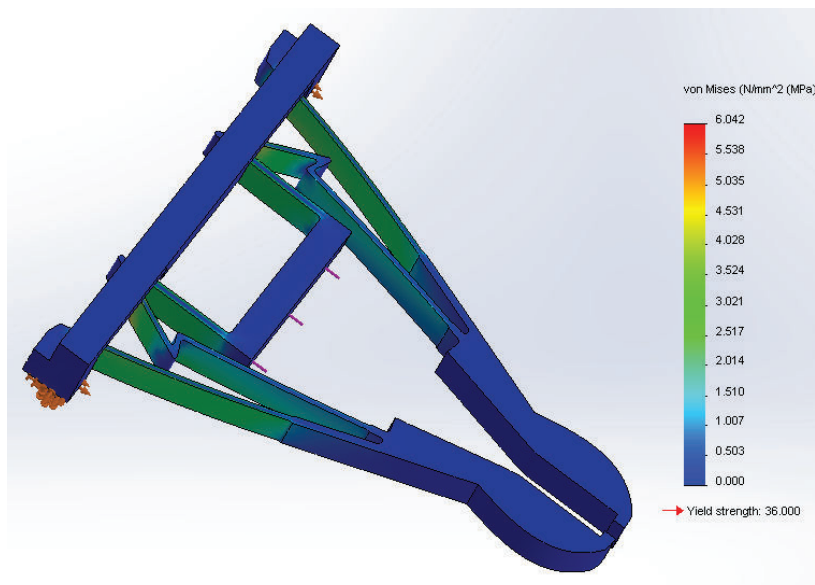


Figure 2.61: Von Mises stresses on the gripper under a 1.3 N pull (no object between claws).

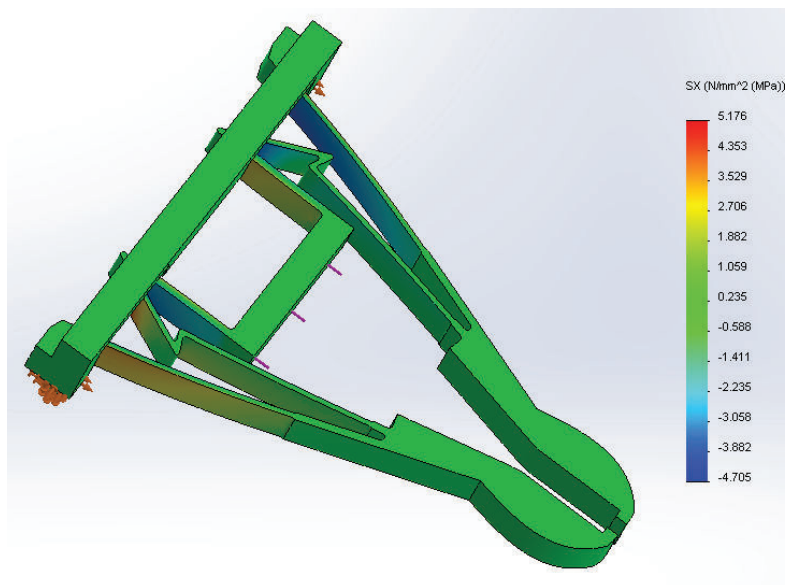


Figure 2.62: X direction stresses on the gripper under a 1.3 N pull (no object between claws).

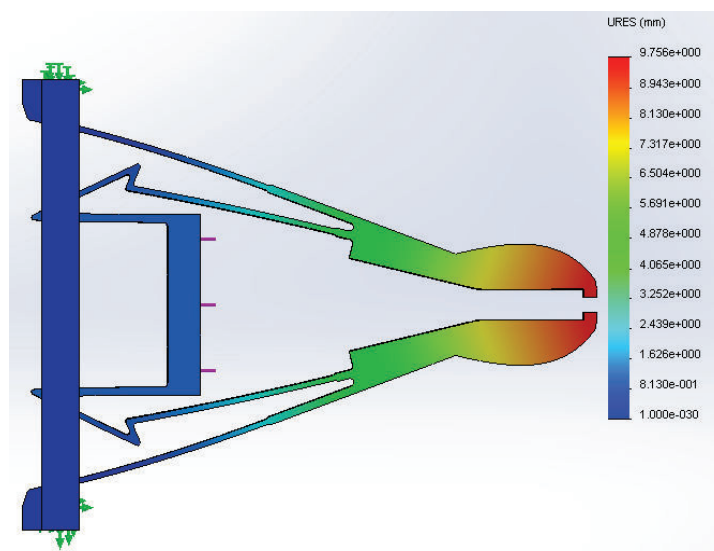


Figure 2.63: Total displacements and final deformation in 1:1 scale for the 1.3N load case.

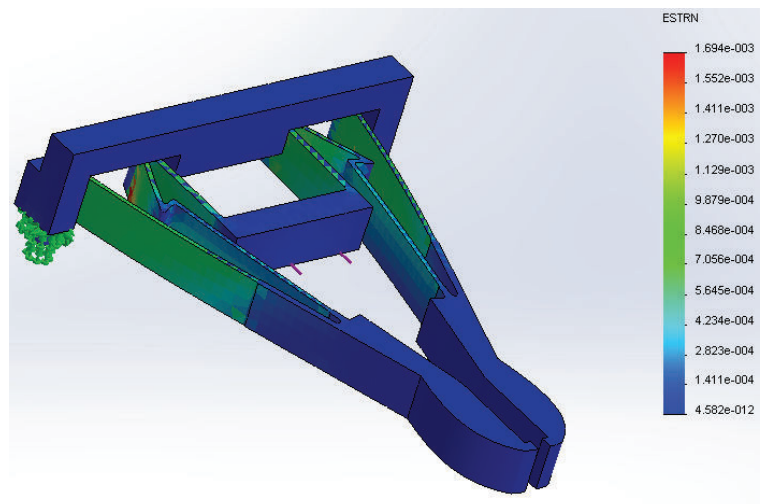


Figure 2.64: Equivalent strain for the 1.3N load case.

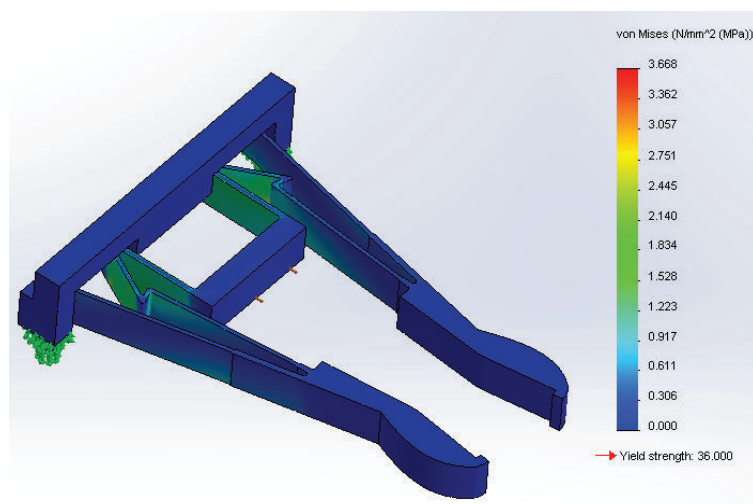


Figure 2.65: Von Mises stresses on the gripper under a 1.3 N pull and with spring connectors to simulate an object between claws.

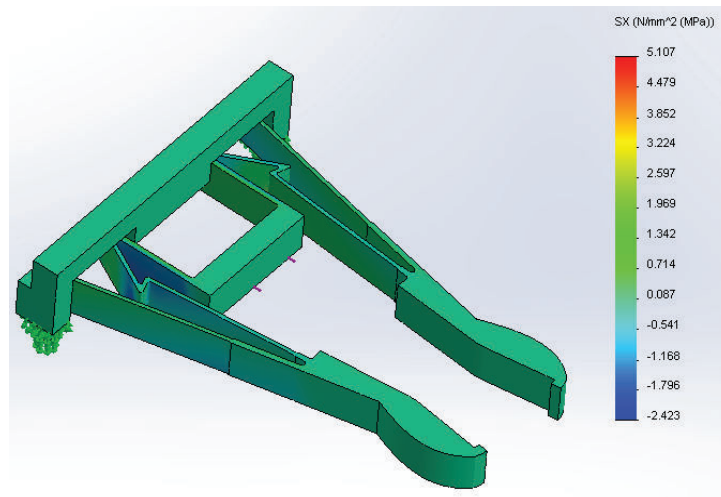


Figure 2.66: X direction stresses on the gripper under a 1.3 N pull and with spring connectors to simulate an object between claws.

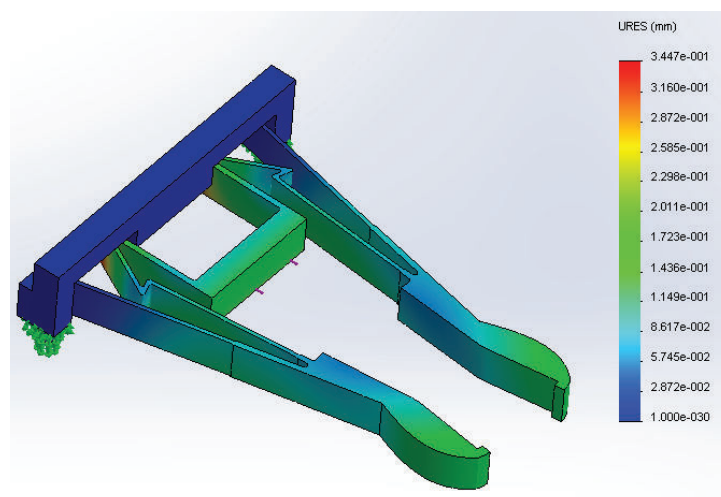


Figure 2.67: Total displacements and final deformation in 1:1 scale for the 1.3N load case with spring connectors between claws.

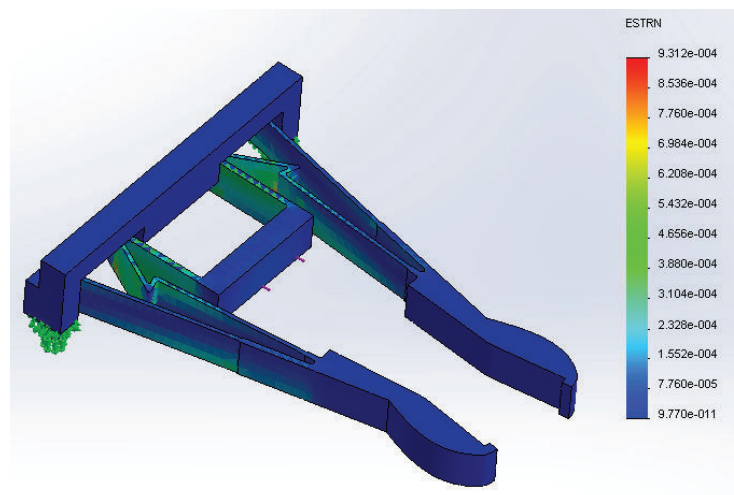


Figure 2.68: Equivalent strain for the 1.3N load case with spring connectors between claws.

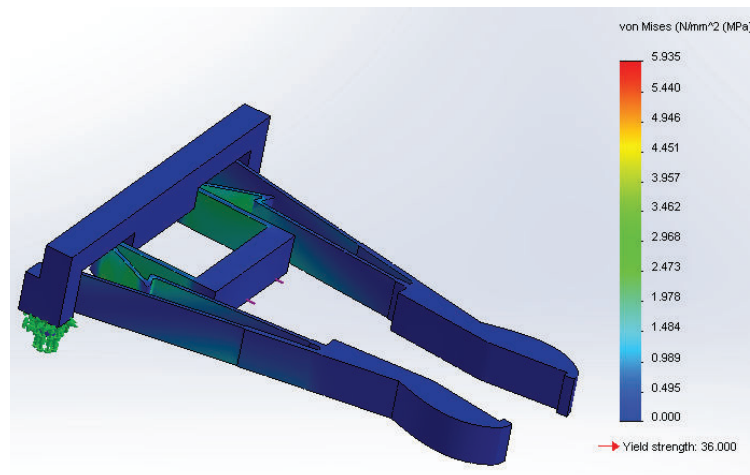


Figure 2.69: Von Mises stresses on the gripper under a 2.1 N pull and with spring connectors to simulate an object between claws.

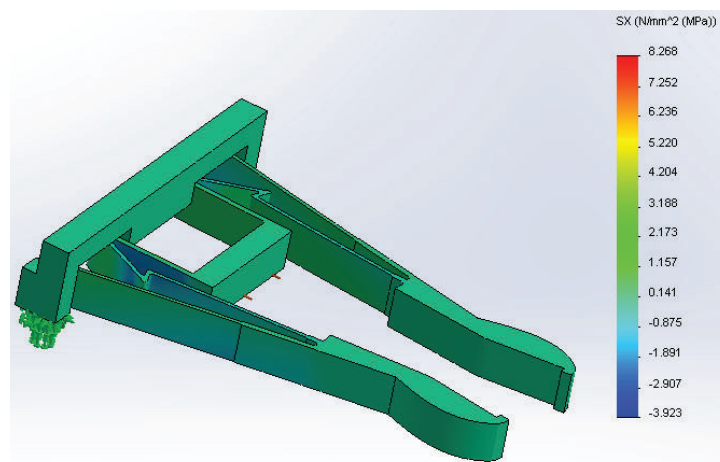


Figure 2.70: X direction stresses on the gripper under a 2.1 N pull and with spring connectors to simulate an object between claws.

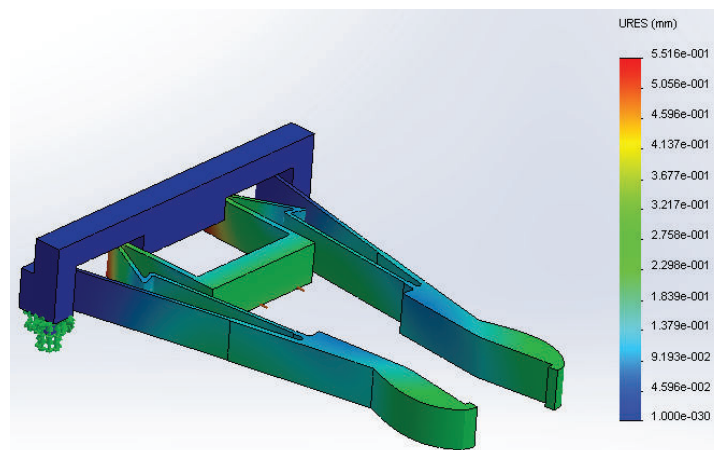


Figure 2.71: Total displacements and final deformation in 1:1 scale for the 2.1 N load case with spring connectors between claws.

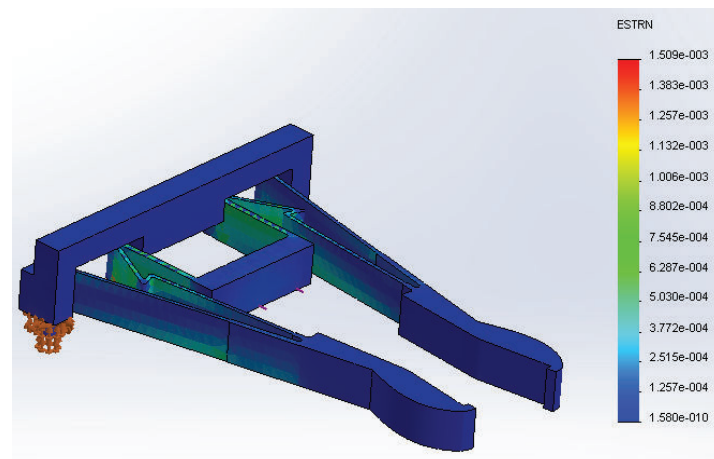


Figure 2.72: Equivalent strain for the 2.1 N load case with spring connectors between claws.

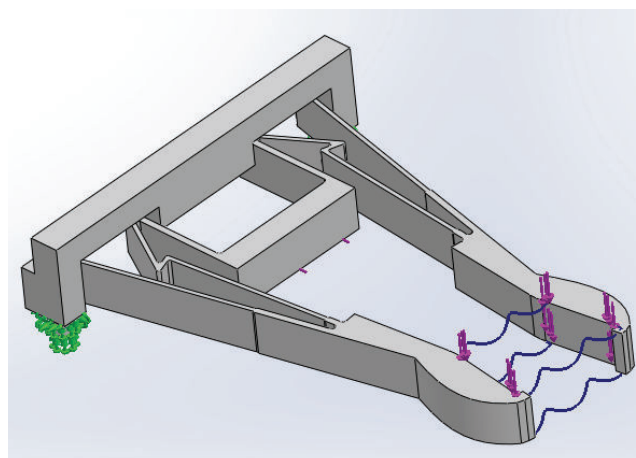


Figure 2.73: Boundary conditions for the case considering out-of-plane deflection.

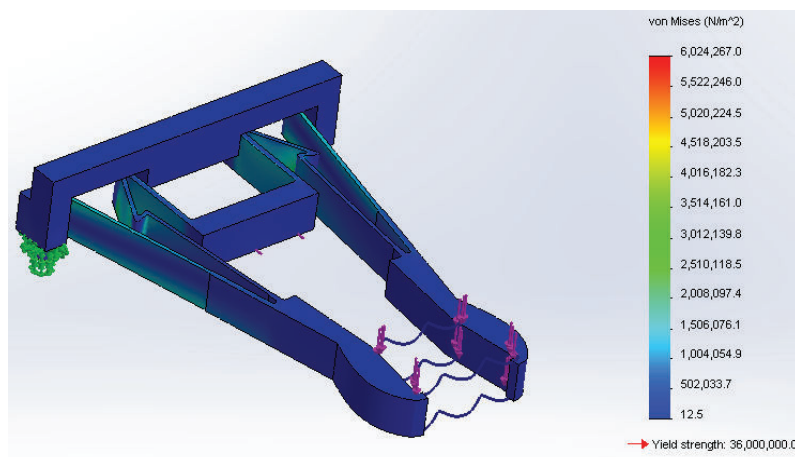


Figure 2.74: Von Mises stresses on the gripper under a 1.3 N pull and with payload weight simulation.

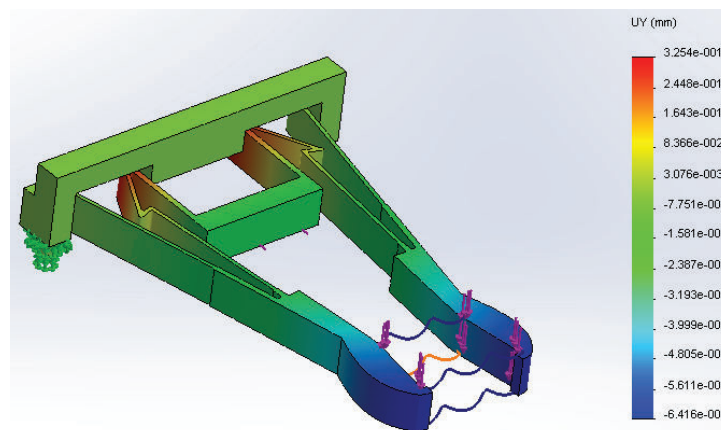


Figure 2.75: Out-of-plane (y) displacements and final deformation in 1:1 scale for the case with payload weight simulation.

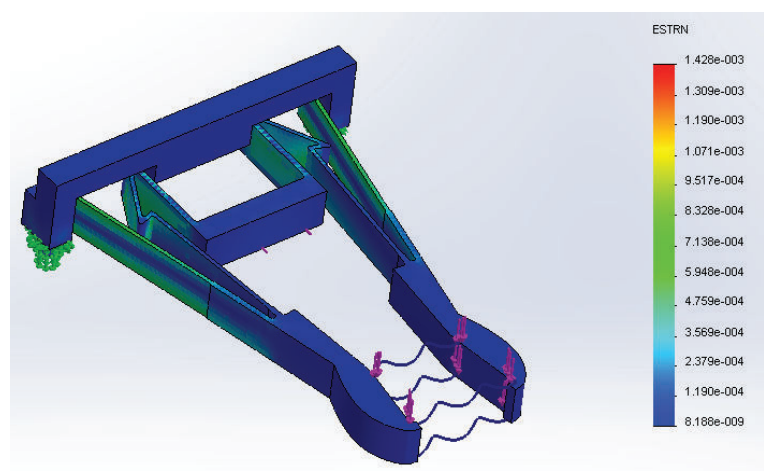


Figure 2.76: Equivalent strain for the 1.3N load case with payload weight simulation.

was created, to identify the best configuration, evaluate the position of the center of gravity, and spot corrections to be implemented in the various components.

When a preferred configuration was frozen (see Figure 2.77), a first physical 3D mock-up was built, using the actual COTS components foreseen for the real μ Rover, and creating all structural parts thanks to a 3D printer. This mock-up was used to check dimensional and layout issues, and to develop, write, and tailor the control code.

A second 3D mock-up was built using a rigid-flex PCB as the rover's chassis.

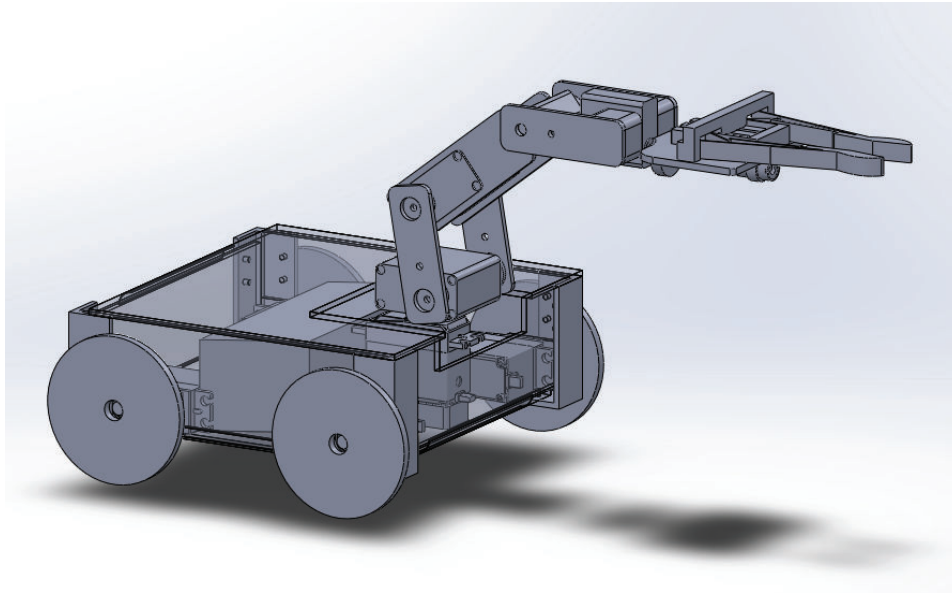


Figure 2.77: Rover 3D model showing all the main components.

The rover's control code is structured on a two-level architecture: a low level program written in C language is resident on the rover's microcontroller, while a high level program written in Matlab is resident on the PC control station. The software on the MCU takes care of low abstraction duties: it opens and handles the communication with the control station, it receives and deciphers commands and actuates them, and sends acknowledgment back to the PC. On the other hand, the software on the control computer is in charge of high abstraction duties: it runs the GUI, checks coherence of user's inputs, computes motor's actuation parameters to perform the requested action, and sends them to the microcontroller onboard the rover.

The actuation of servomotors is carried out thanks to two nested levels, too. The outer level is handled by the control station: it has knowledge of the last speed or position command sent to the servomotors, and of the kinematic model of the rover's arm. Therefore, given a user input, it is able to compute a new set of commands to achieve the new goal. However, this level receives no feedback from the rover, except for an acknowledgment of successful communication. The inner level is handled by the microcontroller and the servos' drivers. The microcontroller translates the command into a PWM signal and sends it to the servo, while the driver interprets the PWM signal and compares it against current servo position or speed, adjusting the output accordingly. In this way the computation and communication

overhead is minimal, even if this means that there is no way of actively correcting errors induced by non-ideal components (backlash in gears, elastic deformation of structural elements, friction...).

Strictly linked to the development of the control code was the realization of a graphical user interface. This task was accomplished thanks to the Matlab GUI development environment (GUIDE). The aim of the GUI is to free the user from the necessity to enter commands in a manual, text-oriented fashion. In fact, a graphical point-and-click oriented interface is much more intuitive, allows a quicker control of the rover and, interposing an abstraction layer between the user and the control subroutines, avoids errors due to wrong input syntax or command sequences.

The GUI dashboard is structured in two sections. The upper section handles the kinematics of the robotic arm. One 3D and three 2D views offer awareness of the arm configuration. Two panels dedicated to the forward and inverse kinematics of the arm allow the user to preview and submit commands. The lower section takes care of the rover's locomotion and of ancillary functions (gripper and laser actuation). On the lower right corner, a message panel explains what is going on and instructs the user on how to correct wrong input that was rejected. In the beginning, the plan was to develop all the control software based on the Matlab Robotics Toolbox [81]. However, after the initial development stages, it was clear that the standard solution routines offered by the toolbox were not slim enough to allow direct control of the robot. Therefore, a completely independent suite of subroutines was written from scratch, starting from a closed form analytical solution of the problem. This resulted in a quicker execution and in the possibility to identify multiple feasible configurations for a given desired target. Details on the mathematical description of the solution will be given at the end of this section. Even if the Robotics Toolbox was not used for the analysis, its graphical functions were simple and well built, therefore the GUI exploits those routines (with minor corrections) to depict the arm configuration.

Regarding the electrical design, the main circuits included in the rover structure are:

- DC/DC conversion and voltage regulation
- Main controller
- Wireless communication module
- Compliant gripper actuation circuit
- Laser control circuit
- Sensors signal conditioning circuit
- Battery meter

Circuits have been designed and simulated with LTSpice, and part of them have been breadboarded to test the performances.

Table 2.32 summarizes the voltage ratings for the various loads, while Table 2.33 describes current and power drain for the same loads. These data led the

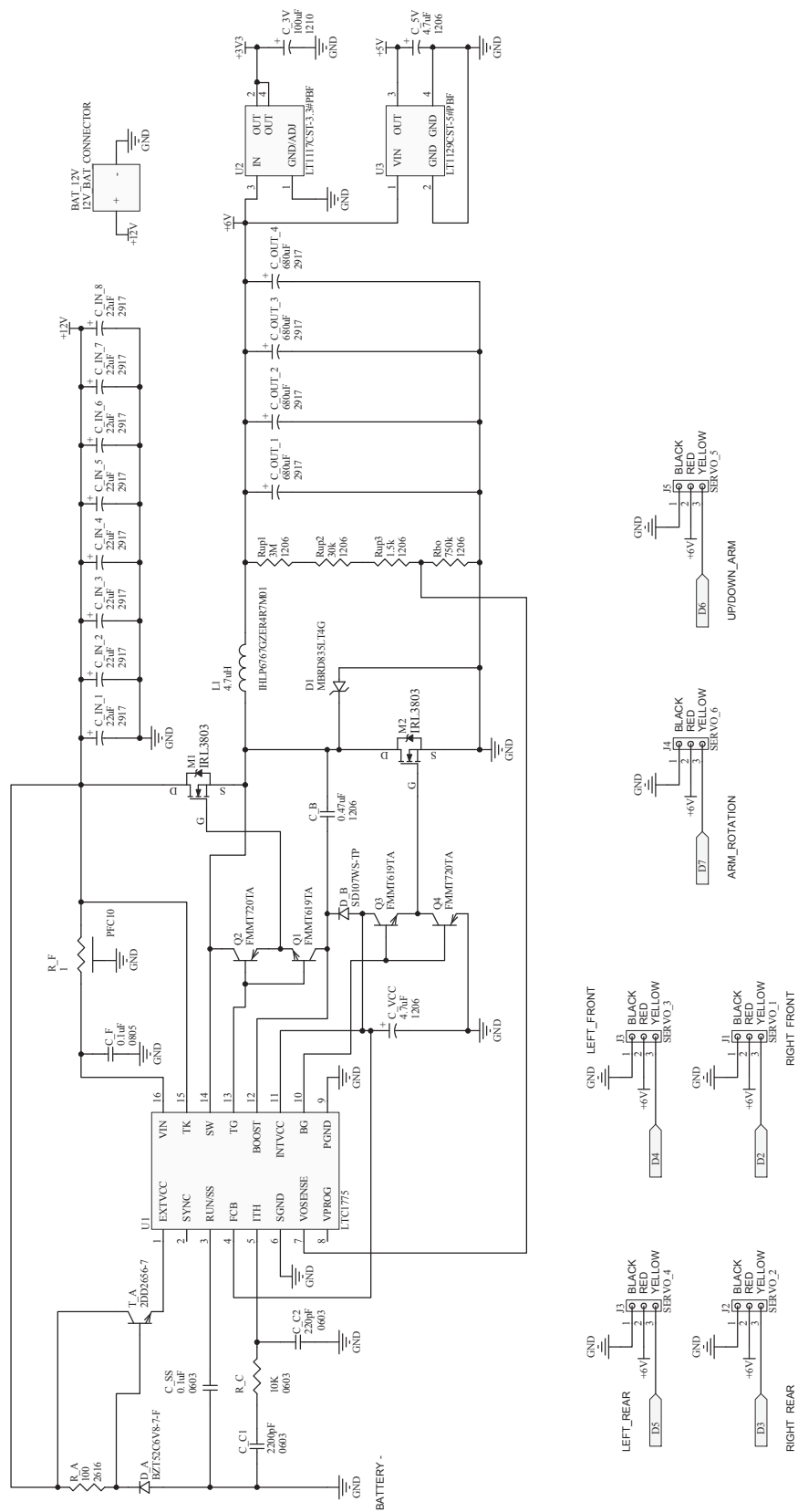


Figure 2.78: Schematic of the DC/DC converter.

Table 2.32: Voltage ratings for various components.

Component	Min (V)	Typ (V)	Max (V)
Servos	4.8	6	7.4
SMA motors	2.5	6	9
Microcontroller	-	5	
RF module	-	3.3	-
Laser module	3	-	6

choice of the battery, which is a 12.0 V Ni-MH battery, with a nominal 2800 mAh capacity and a maximum discharge current of 28 A. It is however clear that it is not possible to use this single unregulated power source to feed all loads; therefore a DC/DC converter unit is needed to regulate level and quality of the power fed to the various elements. The converter of the μ Rover has been designed to have a maximum input voltage of 15 V, a 6/5/3.3 V output and a maximum output current of 20 A. As shown in Figure 2.78, a first switching stage steps down the voltage from 12 V to 6 V, and feeds directly the motors, which can draw a substantial amount of current, and are more tolerant with respect to low quality power. This first stage was designed as a step-down switching regulator mainly for two reasons: first of all because the input-output voltage drop is quite big, and secondly because the loads can draw a substantial current, in the order of several amps. These two considerations, summed together, would make a linear regulator too dissipative and require a substantial heat sink. The first stage is therefore composed by a switching regulator controller that drives external MOSFETs. These MOSFETs have been selected on the basis of their breakdown voltage $V_{BR_{DSS}}$, which had to be greater than 15 V, and of their on-resistance $R_{DS_{ON}}$, which needed to be under $9m\Omega$. Moreover, since the current drain can undergo huge variations in the different operational configurations of the rover, preference has been given to a controller capable of low switching losses at low load currents. In this way the converter's efficiency remains over 90% down to ~ 300 mA and over 80% down to ~ 9 mA. A second stage takes the 6 V input and gives 5 and 3.3 V outputs: the 5 V output is required by all electronic components, while the 3.3 V level is needed by the wireless communication module. All these logic level users require low currents, therefore linear regulators (low dropout, LDO) can be used, receiving the additional benefit of a much better ripple suppression. The most relevant components of the DC/DC converter group are:

- LTC1775 Linear Technology programmable IC buck regulator
- LTC1117 Linear Technology IC LDO 3.3 V regulator
- LTC1129 Linear Technology IC LDO 5 V regulator

Regarding the other components, input and output capacitors have been chosen based on a worst case scenario. For the input ones, the choice has been led by input RMS current (which had to tolerate $I_{O_{max}}/2$), and by a low value of the Equivalent Series Resistance (ESR). On the other hand, the output components have

Table 2.33: Power budget.

Part name	N. of items	Ref. voltage (V)	Current - avg (mA)	Current - spike (mA)	Power - avg (mW)	Power - spike (mW)
Parallax continuous rotation servo	4	6	150	1000	900	-
Parallax standard servo	1	6	150	1000	900	-
HSR-5498SG servo	2	6	200	1200	1200	7200
HS-85BB micro servo	2	4.8	280	-	1344	-
SMA motor	3	4	470	-	1880	-
Laser module	1	6	20	25	120	150
Radio module	1	3.3	55	250	182	825
Microcontroller	1	5	30	-	150	-

been chosen based on the lowest possible ESR, to minimize output voltage ripple. Tantalum capacitors have been chosen, because they have the highest capacitance density. Care has been used in selecting only components that have been surge tested for use in switching power supplies.

Another detail is the external 6V reference for the EXTVcc connection, which is basically obtained through a zener diode that provides a fixed voltage, and a MOSFET, which enables current to be drawn directly from the input. As an alternative, if envelope or complexity considerations require a simplification of the circuit, the EXTVcc line can be connected directly at the 6V output. In that case, care must be used in never changing the components in such a way that the output may become greater than 7V, otherwise this will result in permanent damage to the switching controller.

The details of all components are listed in Table 2.35, that shows the DC/DC converter's Bill of Materials (BoM).

The main controller is based on the open source Arduino architecture, and its role is to coordinate communication and actuation onboard the μ Rover. The principal functional components of the controller are:

- Atmel ATMEGA328P high performance, low power AVR 8-bit microcontroller
- Atmel ATMEGA16U2 AVR 8-bit microcontroller
- CTS ATS16B crystal
- Murata CSTCE16M0V53-R0 16 MHz ceramic resonator

The schematic of the controller section is depicted in Figure 2.79, while its BoM is listed in Table 2.36.

Wireless communication capabilities have been added thanks to XBee wireless modules. One of these small radios is mounted on the rover, and the other is connected to an USB port on the control station. The simple schematic of the circuitry necessary to include the XBee module onboard the rover is represented in Figure 2.80, and the related bill of materials is reported in Table 2.37.

XBees are available in various protocols (ZigBee, XBee, DigiMesh, 802.15.4, 802.11bgn), frequencies (865 MHz, 868 MHz, 900 MHz, 2.4 GHz), and communication ranges (120 m to 4 km, without high gain antenna, up to 80 km with high gain antenna). The one that was chosen for this system is the XBee-PRO Series 1 based on the 802.15.4 networking protocol, and well suited for high-throughput applications requiring low latency and predictable communication timing.

A specific syntax was created for the communication between the rover and the control station. Each message is composed of five bytes: an 8-bit header, three bytes for the content, and an 8-bit terminator. If a received message includes less than five bytes, it is discarded. If the message is valid, the 24-bit content is in turn divided into one byte and one word. The first part of the content represents the microcontroller pin to which the command is addressed, and therefore gives also information regarding which kind of action is required. The remaining 16-bit chunk of the content is a numerical value quantifying the extent of the required action.

Table 2.34: Summary of XBee characteristics.

Characteristic	Value
Power output	63 mW
Indoor or urban range	90 m
Outdoor range	1.6 km RF LOS
RF data rate	250 Kbps
Operating frequency	2.4 GHz
Frequency band	2.4000 - 2.4835 GHz
Networking topology	point-to-point, point-to-multipoint, peer-to-peer
Error handling	retries and acknowledgements
Supply voltage	2.8 - 3.4 VDC
Transmit current	215 mA (@ 3.3 V)
Receive current	55 mA (@ 3.3 V)
Power-down sleep current	<10 μ A at 25°C
Footprint	2.5 cm x 3.3 cm
Weight	3g
Antenna type	chip antenna
Operating temperature	-40°C to 85°C

This message structure, coupled with the intrinsic error handling features of the RF module, gives an highly reliable communication pattern. In two months of testing no failure was detected, and no command was missed or misinterpreted.

All the other circuit patches are depicted in Figure 2.81. They are needed for the following functions: gripper actuation, laser control, pressure sensors signal conditioning, and battery meter. The BoM for all these ancillary circuits are listed in Table 2.38. A relevant point to be noticed in the electrical design is that all parts have common ground. This is necessary to ensure a proper behavior of the system. In particular, the microcontroller section needs to have the same ground as its loads in order to ensure correct actuation, without jitter or noise.

Table 2.35: Bill of materials for the DC/DC converter.

Count	Name	Manufacturer	Package	Reference	Description
Main switching regulator: 15 to 6 V converter					
1	LTC1775CGN#PBF	Linear Technology	16-SSOP	U1	IC REG CTRLR BUCK PWM CM 16-SSOP
2	ECP-U1C104MA5	Panasonic Electronic Components	0805	C_SS, C_F	CAP FILM 0.1UF 16VDC 0805
1	ECH-U1C222JX5	Panasonic Electronic Components	0603	C_C1	CAP FILM 2200PF 16VDC 0603
1	CRCW060310K0JNEAHP	Vishay Dale	0603	R_C	RES 10K OHM .25W 5% 0603 SMD
1	ECH-U1C221GX5	Panasonic Electronic Components	0603	C_C2	CAP FILM 220PF 16VDC 0603
1	PFC10-1RF1	Riedon	PFC10-1RF1	R_F	RES 1 OHM 10W 1% SMD
1	IHLP6767GZER4R7M01	Vishay Dale	17.15mm x 17.15mm	L_1	INDUCTOR POWER 4.7UH 25A SMD
8	TPME226K035R0060	AVX Corporation	2917	C_IN x 8	CAP TANT 22UF 35V 10% 2917
2	IRL3803STRLPBF	International Rectifier	D2Pak	M1, M2	MOSFET N-CH 30V 140A D2PAK
2	FMMT720TA	Diodes Inc	superSOT/SOT-23	Q2, Q4	TRANS SW PNP 40V 1.5A SOT23-3
2	FMMT619TA	Diodes Inc	superSOT/SOT-23	Q1, Q3	TRANS HP NPN 50V 2000MA SOT23-3
1	ECP-U1C474MA5	Panasonic Electronic Components	1206	C_B	CAP FILM 0.47UF 16VDC 1206

Table 2.35: Bill of materials for the DC/DC converter.

Count	Name	Manufacturer	Package	Reference	Description
1	SD107WS-TP	Micro Commercial Co	SOD-323	D_B	DIODE SCHOTTKY 30V 100MA SOD323
1	MBRD835LT4G	ON Semiconductor	DPAK 369C	D_1	DIODE SCHOTTKY 35V 8A DPAK
1	TPSA475K020R1800	AVX Corporation	1206	C_VCC	CAP TANT 4.7UF 20V 10% 1206 low ESR
4	TPME687K006R0018	AVX Corporation	2917	C_OUT x 4	CAP TANT 680UF 6.3V 10% 2917
1	RC1206FR-073ML	Yageo	1206	Rup1	RES 3.00M OHM 1/4W 1% 1206 SMD
1	ERJ-8ENF3002V	Panasonic Electronic Components	1206	Rup2	RES 30.0K OHM 1/4W 1% 1206 SMD
1	ERJ-8ENF1501V	Panasonic Electronic Components	1206	Rup3	RES 1.50K OHM 1/4W 1% 1206 SMD
1	ERJ-8ENF7503V	Panasonic Electronic Components	1206	Rbo	RES 750K OHM 1/4W 1% 1206 SMD
6 V to 3.3 V converter					
1	LT1117CST-3.3#PBF	Linear Technology	SOT-223	U3	IC REG LDO 3.3V .8A SOT-223
1	TPSB107K006R0400	AVX Corporation	1210	C_3V	CAP TANT 100UF 6.3V 10% 1210
6 V to 5 V converter					
1	LT1129CST-5#PBF	Linear Technology	SOT-223-3	U2	IC REG LDO 5V .7A SOT223-3

Table 2.35: Bill of materials for the DC/DC converter.

Count	Name	Manufacturer	Package	Reference	Description
1	TPSA475M010R1400	AVX Corporation	1206	C_5V	CAP TANT 4.7UF 10V 20% 1206
EXTVcc reference					
1	BZT52C6V8-7-F	Diodes Inc	SOD-123	Da	DIODE ZENER 6.8V 500MW SOD-123
1	1879011-5	TE Connectivity	2616	Ra	RES 100 OHM 2W 5% SMD
1	2DD2656-7	Diodes Inc	SOT-323	Ta	transistor

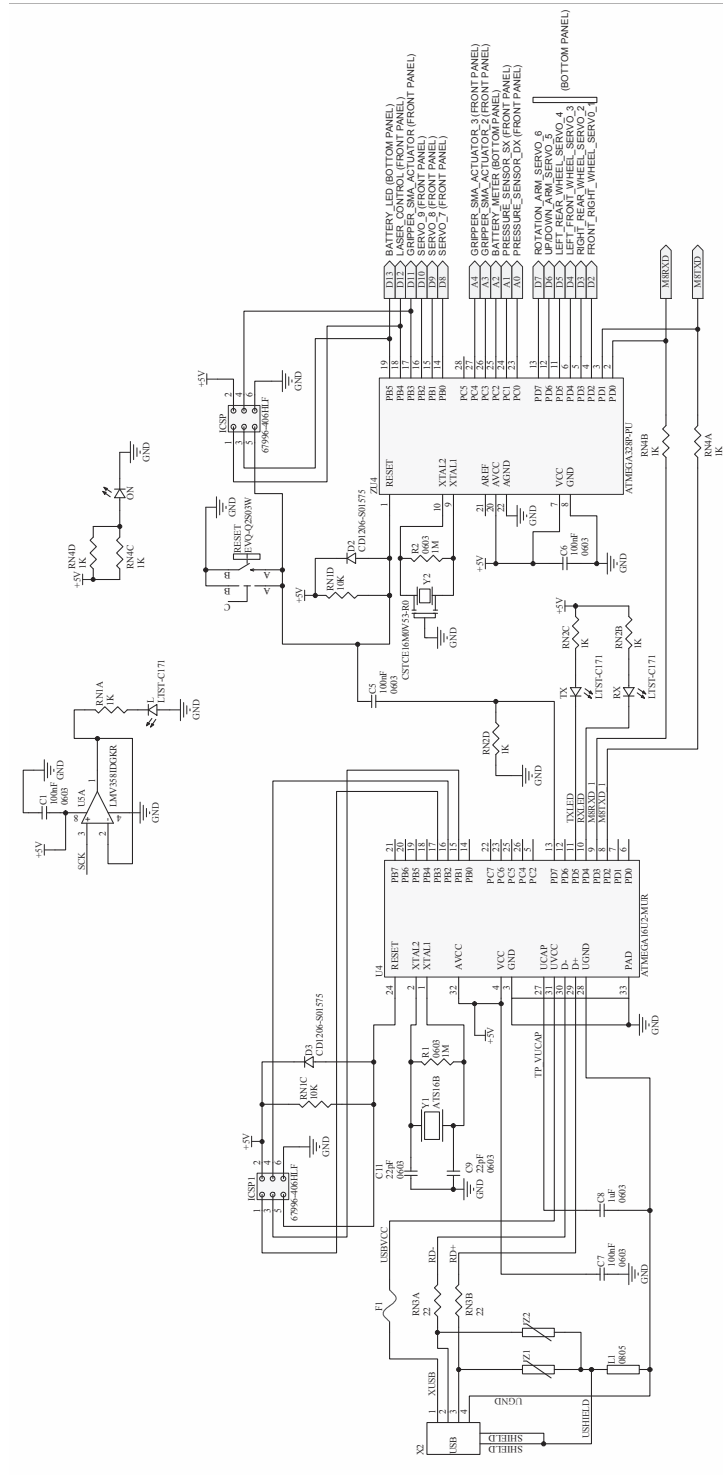


Figure 2.79: Schematic of the controller section.

Table 2.36: Bill of materials for the controller.

Count	Name	Manufacturer	Package	Reference	Description
2	500R14N220JV4T	Johanson	0603	C9, C11	22pF capacitor ceramic
6	CC0603KRX7R9BB104	Yageo	0603	C1, C2, C4, C5, C6, C7	100nF capacitor ceramic
2	GRM188R60J105KA01D	Murata	0603	C3, C8	1uF capacitor ceramic 6.3V
2	EMVA250ADA470MF55G	UCC	SMD	PC1, PC2	47uF electrolytic capacitor SMT
1	USB-B1HSW6	On Shore	PTH	X2	USB type B connector right-angle - white
1	ATS16B	CTS	HC49/US	Y1	16MHz 18pF crystal
2	CD1206-S01575	Bourns	1206	D2, D3	Switching diode
1	S2M-13-F	Diodes Inc	SMB	D1	Switching diode
1	BLM21PG221SN1D	Murata	0805	L1	Ferrite 50mOhm@DC / 220ohm@100MHz
1	MF-MSMF050-2	Bourns	1812	F1	Fuse 500mA 15V
1	ATMEGA16U2-MUR	Atmel	VQFN-32	U3	ATmega16U2 16kB FLASH
1	ATMEGA328P-PU	Atmel	DIP-28	ZU4	ATmega328P SMD
1	LMV358IDGKR	TI	MSOP-8	U5	Dual op-amp RRO
1	LP2985-33DBVR	TI	SOT-23-5	U2	150mA 3.3V LDO
1	NCP1117ST50T3G	ON Semi	SOT-223	U1	1A 5V LDO
1	LG R971-KN-1-0-20-R18	Osram	0805	ON	LED green
3	LTST-C171KSKT	Lite-On	0805	L, RX, TX	LED yellow
1	CAY16-220J4LF	Bourns	1206	RN3	22ohm 4-resistor series array
2	CAY16-102J4LF	Bourns	1206	RN2, RN4	1k 4-resistor series array
1	CAY16-103J4LF	Bourns	1206	RN1	10k 4-resistor series array

Table 2.36: Bill of materials for the controller.

Count	Name	Manufacturer	Package	Reference	Description
1	ERJ-3GEYJ105V	Panasonic	0603	R1, R2	1M resistor 5% 1/10W
1	CSTCE16M0V53-R0	Murata	SMD	Y2	16MHz resonator with built-in caps
1	EVQ-Q2S03W	Panasonic	SMT	RESET	OFF-Mom tactile switch
1	FDN340P	Fairchild	SSOT-3	T1	20V P-type MOSFET
2	CG0603MLC-05E	Bourns	0603	Z1, Z2	Low-capacitance varistor 5V WV

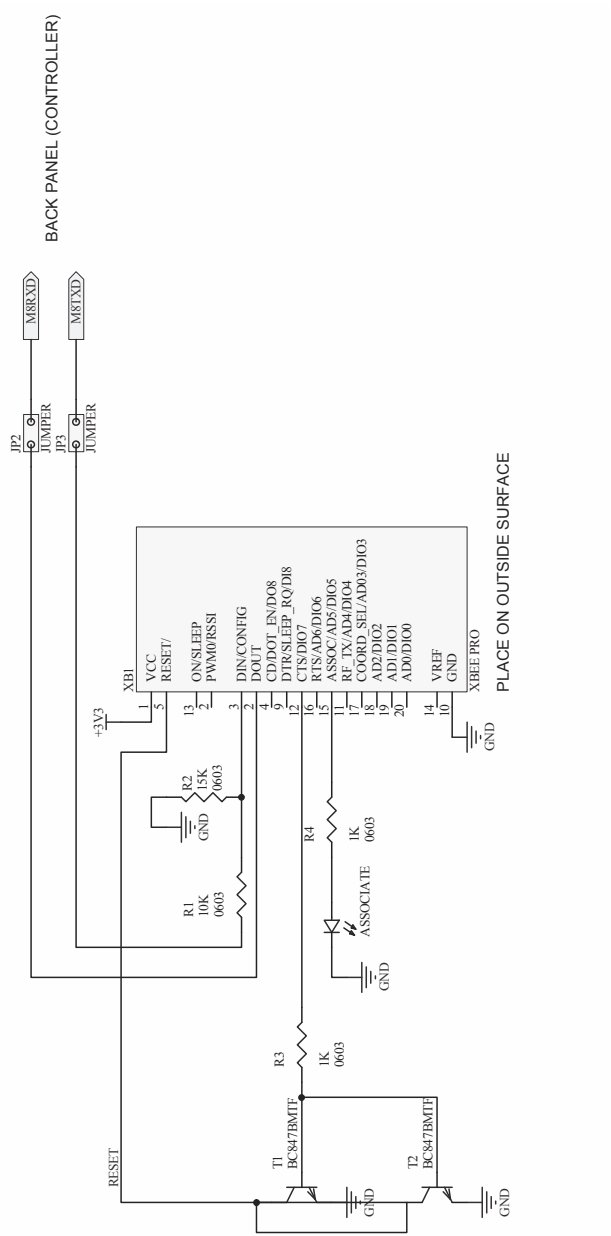


Figure 2.80: Schematic of the XBee section.

Table 2.37: Bill of materials for the RF module section.

Count	Name	Manufacturer	Package	Reference	Description
1	SN74LVC1G125DCKR	Texas Instruments	SC-70-5		IC BUS BUFF TRI-ST N-INV
1	CC0603KRX7R9BB104	Yageo	0603	C1	CAP CER 0.1UF 50V 10% X7R 0603
2	M22-7131042	Harwin Inc	2mm pitch	JP1, JP9	pin header 10x1 F p2.0mm
1	CRCW060310K0JNEAHP	Vishay Dale	0603	R1	RES 10K OHM .25W 5% 0603 SMD
2	T520B106M016ATE100	Kemet	1411	C2, C3	CAP TANT 10UF 16V 20% 1411 Polymer
1	CRCW060315K0FKEAHP	Vishay Dale	0603	R2	RES 15.0K OHM .25W 1% 0603 SMD
2	CRCW06031K00FKEAHP	Vishay Dale	0603	R3, R4	RES 1.00K OHM .25W 1% 0603 SMD
1	LG R971-KN-1-0-20-R18	OSRAM Opto Semiconductors Inc	0805	ASSOCIATE	LED CHIPLEDED 570NM GREEN 0805 SMD
1	EVQ-Q2S03W	Panasonic		S1	OFF-Mom tactile switch
1	BC847BMTF	Fairchild Semiconductor	SOT-23	T1	BC847B TRANSISTOR NPN 45V 0.1A SOT23

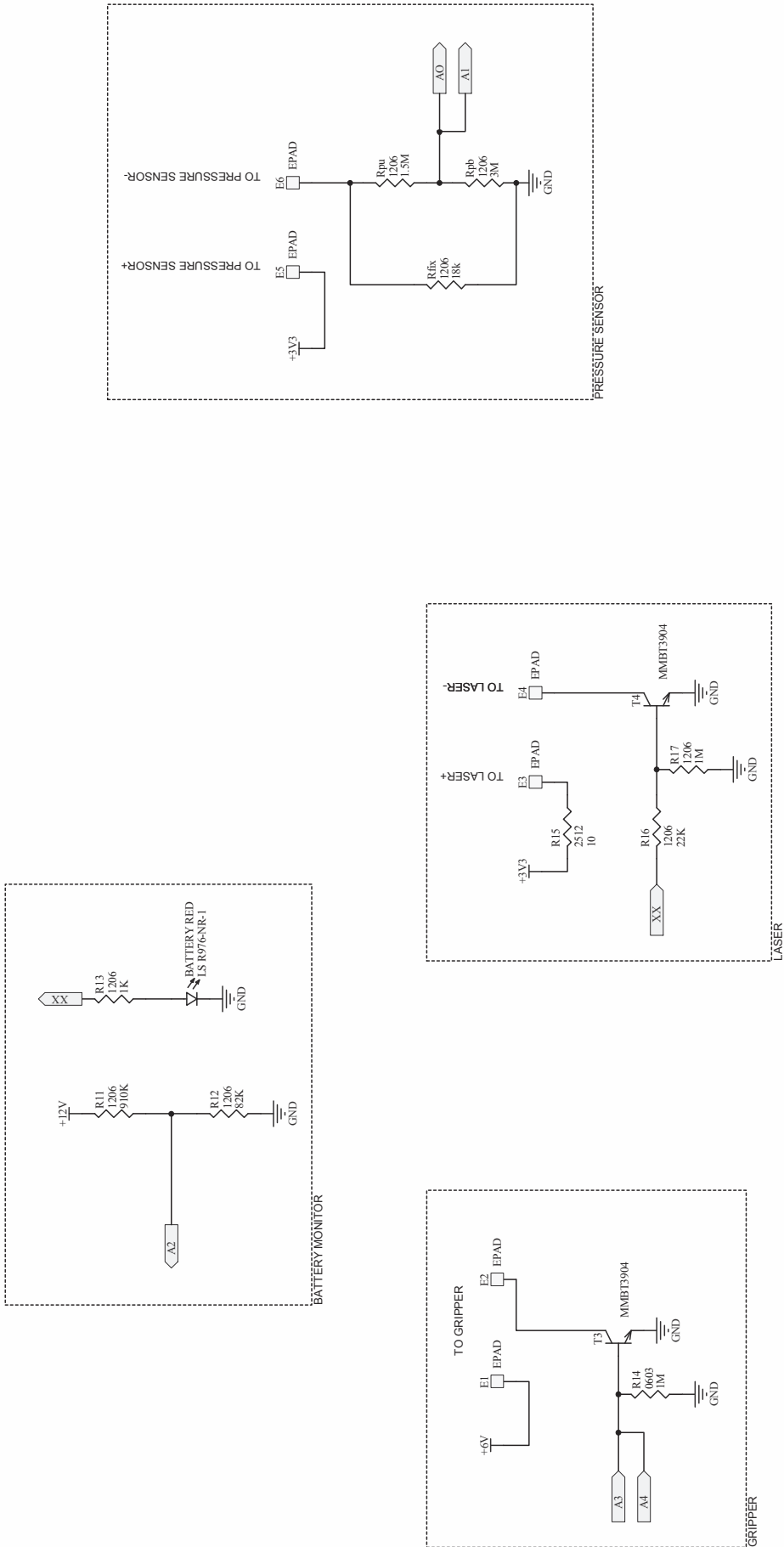


Figure 2.81: Schematics of miscellaneous circuits.

Table 2.38: Bill of materials for all accessory circuitry.

Laser control				
Count	Name	Manufacturer	Package	Reference
1	MMBT3904	Fairchild Semiconductor	SOT-23	T1
1	1622820-4	TE Connectivity	2512	R3
1	ERJ-8GEYJ105V	Panasonic Electronic Components	1206	R2
1	ERJ-8ENF2202V	Panasonic Electronic Components	1206	R1

Battery meter				
Count	Name	Manufacturer	Package	Reference
1	ERJ-8GEYJ914V	Panasonic Electronic Components	1206	R1
1	ERJ-8GEYJ823V	Panasonic Electronic Components	1206	R2
1	CR1206-FX-1001ELF	Bourns Inc.	1206	R3
1	LS R976-NR-1	OSRAM Opto Semiconductors Inc	0805	BATTERY RED

RES 910K OHM 1/4W 5%
1206 SMD

RES 82K OHM 1/4W 5%
1206 SMD

RES 1.0K OHM 1/4W 1%
1206 SMD

LED CHIPLED 633NM
RED 0805 SMD

Table 2.38: Bill of materials for all accessory circuitry.

Pressure detection					
Count	Name	Manufacturer	Package	Reference	Description
1	ERJ-8ENF1802V	Panasonic Electronic Components	1206	Rfix	RES 18.0K OHM 1/4W 1% 1206 SMD
1	ERJ-8ENF1504V	Panasonic Electronic Components	1206	Rpu	RESISTOR 1.50M OHM 1/4W 1% 1206
1	RC1206FR-073ML	Yageo	1206	Rpb	RES 3.00M OHM 1/4W 1% 1206 SMD

Another relevant part of the research activity consisted in studying the kinematics of the rover's manipulator arm.

As any other industrial robot, the arm of the rover is made up of links and joints. The links are the individual rigid bodies that make up the arm, rigid members that may have relative motion with respect to other links¹². The connection between two links is made through a joint, the component that allows relative motion. This relative motion can be expressed by a single coordinate, and defines the kind of joint: revolute (i.e. rotary) or prismatic (translatory).

In general, a robot with n joints will have $n + 1$ links. These links are numbered from 0, which is the immobile grounded base link, up to n , which represents the end-effector link. On the other hand, numbering of joints starts from 1, for the joint connecting the first movable link to the base link, and increases up to n .

The rover's arm is made of six links, numbered from 0 to 5. The zero-th and first link correspond to the body of the base motor (the one that gives the "waist" rotation) and the body of the first articulation motor (the "shoulder"), respectively. The second link is the structure segment that connects the first articulation motor to the second one (the "elbow"). In a similar manner the third link connects the "elbow" to the third articulation servo ("wrist pitch"), and the fourth link connects the "wrist pitch" to the "wrist roll" articulation servo. The last link, number five, is the end effector.

Each link i is connected to its lower link $i - 1$ at its proximal end by joint i and is connected to its upper link $i + 1$ at its distal end by joint $i + 1$. Figure 2.82 shows how the numbering scheme is applied to the rover's arm.

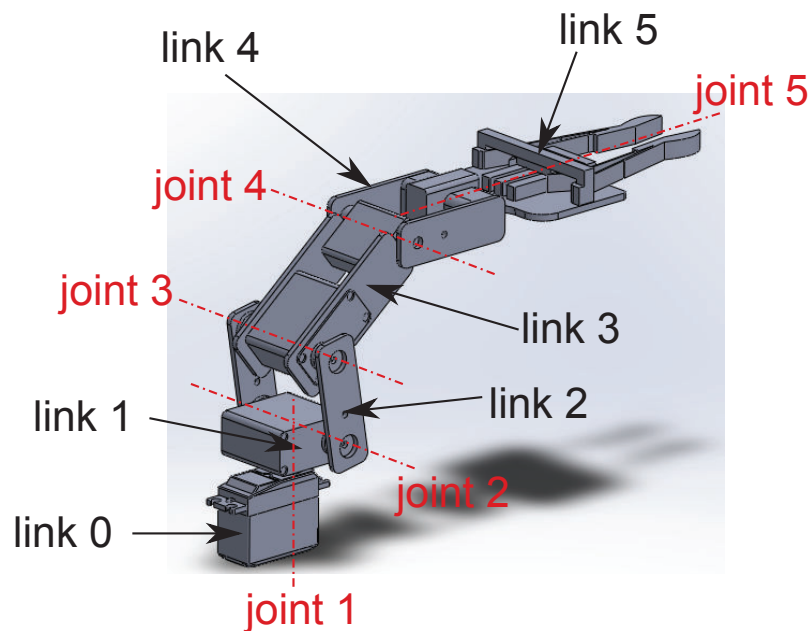


Figure 2.82: Nomenclature and numbering of arm's links and joints. Link 0 is the fixed base. Joint 1 is the waist articulation, joint 2 is the shoulder, joint 3 is the elbow, joint 4 is the pitch DoF of the wrist, and joint 5 is the roll DoF of the wrist.

¹²For this reason, if two or more members are distinct geometrical entities, but are connected together such that no relative motion can occur among them, they are considered as a single link.

The problem of describing the arm's kinematics can be stated as the problem of attaching a local mobile coordinate frame and a set of geometric parameters to each link-joint couple, and describing the orientation of each local coordinate frame with respect to the global fixed coordinate frame which coincides with the fixed base frame.

In general, local coordinate frames B_i can be defined in many different ways, but in 1955 Jacques Denavit and Richard Hartenberg introduced a successful convention to standardize the coordinate frames for spatial linkages. That convention is therefore called Denavit-Hartenberg (DH or D-H) method.

According to Denavit-Hartenberg notation, all joints, without exception, are represented by a z axis, which coincide with the movement axis. In particular, the z_i axis is aligned with the $i + 1$ joint axis. Then the x_i axis can be defined along the common normal between the z_{i-1} and z_i axes, pointing from the z_{i-1} to the z_i axes. The y_i axis is simply obtained with the right-hand rule, but is usually neglected from graphical representations for the sake of clarity.

Figure 2.83 shows how coordinate frames are set for the rover's arm.

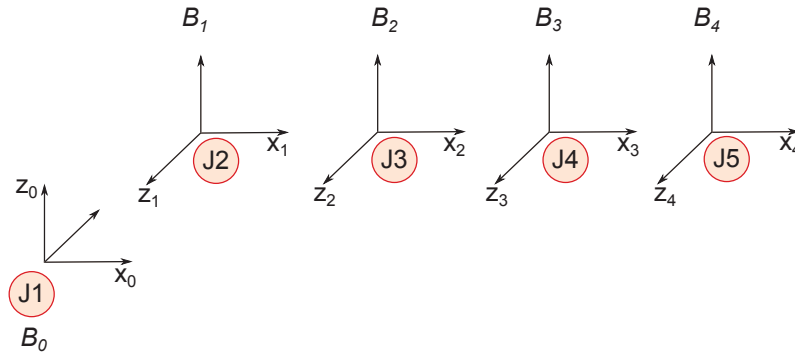


Figure 2.83: Coordinate frames of arm's links and joints. Since joint 5 is not included in the inverse kinematics study, the orientation of the z axis is not coincident with the movement axis.

As mentioned before, a DH coordinate frame can be identified by certain geometrical parameters. These parameters are called a_i , α_i , θ_i , and d_i .

The a_i quantity, called *link length*, is the kinematic length of link i and is the distance between z_{i-1} and z_i axes along the x_i axis.

The α_i quantity, called *link twist*, is the required rotation of the z_{i-1} axis about the x_i axis to become parallel to the z_i axis.

The d_i quantity, called *joint distance* or *link offset*, is the distance between x_{i-1} and x_i axes along the z_{i-1} axis.

The θ_i quantity, called *joint angle*, is the required rotation of x_{i-1} axis about the z_{i-1} axis to become parallel to the x_i axis.

The parameters θ_i and d_i are called joint parameters, because they define the extent of the motion that takes place at joint i . In particular, for a revolute (R) joint the value of d_i is fixed, while θ_i is the joint variable, and on the contrary, for a prismatic (P) joint the value of θ_i is fixed and d_i is the only joint variable.

It is possible to describe the position and orientation of each coordinate frame with respect to the previous and following ones by means of the rigid motion quantities, i.e. a translation and a rotation. In particular, the position of the moving

origin relative to the fixed origin is given by a translation vector t (3×1), and the rotation of the moving frame relative to the fixed frame is given by a rotation matrix R (3×3). To express relative motion more easily, however, it is possible to introduce a homogeneous transformation T (4×4), which includes both of the previous quantities:

$$T = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix}.$$

Since the geometry and movement are completely defined by the four DH parameters, it is possible to demonstrate ([80]) that the homogeneous transformation matrix that converts the position of a point from the i -th reference frame to the $(i - 1)$ -th reference frame is the following:

$${}^{i-1}T_i = \begin{bmatrix} \cos\theta_i & -\sin\theta_i\cos\alpha_i & \sin\theta_i\sin\alpha_i & a_i\cos\theta_i \\ \sin\theta_i & \cos\theta_i\cos\alpha_i & -\cos\theta_i\sin\alpha_i & a_i\sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Once the transformation matrix has been written for all links, it is easy to express the position of a point in the end-effector (n) coordinate frame with respect to the fixed global frame (0). All that it takes is a sequence of matrix multiplications:

$${}^0T_n = {}^0T_1 {}^1T_2 \dots {}^{n-2}T_{n-1} {}^{n-1}T_n.$$

Each of the homogeneous transformation matrices ${}^{i-1}T_i$ is function of the respective joint parameter, therefore it is possible to identify two problems for the control of a robotic arm: a forward kinematics and an inverse kinematics problem. The forward kinematics problem consists in identifying the position in the fixed global coordinate frame of a point attached to the end-effector coordinate frame, with all joint variables given. The inverse kinematics problem consists in identifying a feasible set of joint variables needed to bring a point attached to the end-effector coordinate frame to a desired position in the fixed global reference frame. Both problems were faced for the control of the μ Rover manipulator arm.

The decision was made to give the end-user direct control over the wrist-roll degree of freedom, and therefore to decouple it from all other DoFs. Because of this assumption, the kinematic description of the arm was left with four degrees of freedom, i.e. four joint variables: the base or waist angle θ_1 , the shoulder angle θ_2 , the elbow angle θ_3 , and the wrist-pitch angle θ_4 . The arm can be described by the set of Denavit-Hartenberg parameters summarized in Table 2.39.

Moreover, it must be considered that each joint variable is constrained to the interval $[-\pi/2, \pi/2]$ because of physical limitations of the servo motors.

Therefore the homogeneous transformation matrices for the four links are the following:

$${}^0T_1 = \begin{bmatrix} \cos\theta_1 & 0 & \sin\theta_1 & a_1\cos\theta_1 \\ \sin\theta_1 & 0 & -\cos\theta_1 & a_1\sin\theta_1 \\ 0 & 1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Table 2.39: Denavit-Hartenberg parameters for the manipulator arm of the μ Rover (rest configuration with all joint variables equal to zero).

Link	a_i (mm)	α_i (rad)	d_i (mm)	θ_i (rad)
1	10	$\pi/2$	10	0
2	40	0	0	0
3	69	0	0	0
4	57	0	0	0

$${}^1T_2 = \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 & a_2\cos\theta_2 \\ \sin\theta_2 & \cos\theta_2 & 0 & a_2\sin\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^2T_3 = \begin{bmatrix} \cos\theta_3 & -\sin\theta_3 & 0 & a_3\cos\theta_3 \\ \sin\theta_3 & \cos\theta_3 & 0 & a_3\sin\theta_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^3T_4 = \begin{bmatrix} \cos\theta_4 & -\sin\theta_4 & 0 & a_4\cos\theta_4 \\ \sin\theta_4 & \cos\theta_4 & 0 & a_4\sin\theta_4 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

As a consequence, the global transformation matrix ${}^0T_4 = {}^0T_1 {}^1T_2 {}^2T_3 {}^3T_4$ results in:

$${}^0T_4 = \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \\ m_{41} & m_{42} & m_{43} & m_{44} \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & x \\ r_{21} & r_{22} & r_{23} & y \\ r_{31} & r_{32} & r_{33} & z \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

where the upper left 3 x 3 matrix describes the rotation of the end reference frame with respect to the global reference frame, while the first three rows of the last column represent the translation of the origin of the end coordinate frame with

respect to the global coordinate frame. In particular:

$$\begin{aligned}
m_{11} &= (-\cos\theta_1 \cos\theta_2 \sin\theta_3 - \cos\theta_1 \sin\theta_2 \cos\theta_3) \sin\theta_4 + \\
&\quad + (\cos\theta_1 \cos\theta_2 \cos\theta_3 - \cos\theta_1 \sin\theta_2 \sin\theta_3) \cos\theta_4 \\
m_{12} &= (\cos\theta_1 \sin\theta_2 \sin\theta_3 - \cos\theta_1 \cos\theta_2 \cos\theta_3) \sin\theta_4 + \\
&\quad + (-\cos\theta_1 \cos\theta_2 \sin\theta_3 - \cos\theta_1 \sin\theta_2 \cos\theta_3) \cos\theta_4 \\
m_{13} &= \sin\theta_1 \\
m_{14} &= (-a_4 \cos\theta_1 \cos\theta_2 \sin\theta_3 - a_4 \cos\theta_1 \sin\theta_2 \cos\theta_3) \sin\theta_4 + \\
&\quad + (a_4 \cos\theta_1 \cos\theta_2 \cos\theta_3 - a_4 \cos\theta_1 \sin\theta_2 \sin\theta_3) \cos\theta_4 + \\
&\quad - a_3 \cos\theta_1 \sin\theta_2 \sin\theta_3 + a_3 \cos\theta_1 \cos\theta_2 \cos\theta_3 + a_2 \cos\theta_1 \cos\theta_2 + a_1 \cos\theta_1 \\
m_{21} &= (-\sin\theta_1 \cos\theta_2 \sin\theta_3 - \sin\theta_1 \sin\theta_2 \cos\theta_3) \sin\theta_4 + \\
&\quad + (\sin\theta_1 \cos\theta_2 \cos\theta_3 - \sin\theta_1 \sin\theta_2 \sin\theta_3) \cos\theta_4 \\
m_{22} &= (\sin\theta_1 \sin\theta_2 \sin\theta_3 - \sin\theta_1 \cos\theta_2 \cos\theta_3) \sin\theta_4 + \\
&\quad + (-\sin\theta_1 \cos\theta_2 \sin\theta_3 - \sin\theta_1 \sin\theta_2 \cos\theta_3) \cos\theta_4 \\
m_{23} &= -\cos\theta_1 \\
m_{24} &= (-a_4 \sin\theta_1 \cos\theta_2 \sin\theta_3 - a_4 \sin\theta_1 \sin\theta_2 \cos\theta_3) \sin\theta_4 + \\
&\quad + (a_4 \sin\theta_1 \cos\theta_2 \cos\theta_3 - a_4 \sin\theta_1 \sin\theta_2 \sin\theta_3) \cos\theta_4 + \\
&\quad - a_3 \sin\theta_1 \sin\theta_2 \sin\theta_3 + a_3 \sin\theta_1 \cos\theta_2 \cos\theta_3 + a_2 \sin\theta_1 \cos\theta_2 + a_1 \sin\theta_1 \\
m_{31} &= (\cos\theta_2 \cos\theta_3 - \sin\theta_2 \sin\theta_3) \sin\theta_4 + \\
&\quad + (\cos\theta_2 \sin\theta_3 + \sin\theta_2 \cos\theta_3) \cos\theta_4 \\
m_{32} &= (-\cos\theta_2 \sin\theta_3 - \sin\theta_2 \cos\theta_3) \sin\theta_4 + \\
&\quad + (\cos\theta_2 \cos\theta_3 - \sin\theta_2 \sin\theta_3) \cos\theta_4 \\
m_{33} &= 0 \\
m_{34} &= (a_4 \cos\theta_2 \cos\theta_3 - a_4 \sin\theta_2 \sin\theta_3) \sin\theta_4 + \\
&\quad + (a_4 \cos\theta_2 \sin\theta_3 + a_4 \sin\theta_2 \cos\theta_3) \cos\theta_4 + \\
&\quad + a_3 \cos\theta_2 \sin\theta_3 + a_3 \sin\theta_2 \cos\theta_3 + a_2 \sin\theta_2 + d_1 \\
m_{41} &= 0 \\
m_{42} &= 0 \\
m_{43} &= 0 \\
m_{44} &= 1
\end{aligned}$$

Through trigonometric simplification, it is possible to rewrite the matrix entries as:

$$\begin{aligned}
m_{11} &= \cos\theta_1 \cos(\theta_3 + \theta_2) \cos\theta_4 - \cos\theta_1 \sin(\theta_3 + \theta_2) \sin\theta_4 \\
m_{12} &= -\cos\theta_1 \cos(\theta_3 + \theta_2) \sin\theta_4 - \cos\theta_1 \sin(\theta_3 + \theta_2) \cos\theta_4 \\
m_{13} &= \sin\theta_1 \\
m_{14} &= -a_4 \cos\theta_1 \sin(\theta_3 + \theta_2) \sin\theta_4 + a_4 \cos\theta_1 \cos(\theta_3 + \theta_2) \cos\theta_4 + \\
&\quad + a_3 \cos\theta_1 \cos(\theta_3 + \theta_2) + a_2 \cos\theta_1 \cos(\theta_2) + a_1 \cos\theta_1 \\
m_{21} &= \sin\theta_1 \cos(\theta_3 + \theta_2) \cos\theta_4 - \sin\theta_1 \sin(\theta_3 + \theta_2) \sin\theta_4 \\
m_{22} &= -\sin\theta_1 \cos(\theta_3 + \theta_2) \sin\theta_4 - \sin\theta_1 \sin(\theta_3 + \theta_2) \cos\theta_4 \\
m_{23} &= -\cos\theta_1 \\
m_{24} &= -a_4 \sin\theta_1 \sin(\theta_3 + \theta_2) \sin\theta_4 + a_4 \sin\theta_1 \cos(\theta_3 + \theta_2) \cos\theta_4 + \\
&\quad + a_3 \sin\theta_1 \cos(\theta_3 + \theta_2) + a_2 \sin\theta_1 \cos\theta_2 + a_1 \sin\theta_1 \\
m_{31} &= \cos(\theta_3 + \theta_2) \sin\theta_4 + \sin(\theta_3 + \theta_2) \cos\theta_4 \\
m_{32} &= \cos(\theta_3 + \theta_2) \cos\theta_4 - \sin(\theta_3 + \theta_2) \sin\theta_4 \\
m_{33} &= 0 \\
m_{34} &= a_4 \cos(\theta_3 + \theta_2) \sin\theta_4 + a_4 \sin(\theta_3 + \theta_2) \cos\theta_4 + \\
&\quad + a_3 \sin(\theta_3 + \theta_2) + a_2 \sin\theta_2 + d_1 \\
m_{41} &= 0 \\
m_{42} &= 0 \\
m_{43} &= 0 \\
m_{44} &= 1
\end{aligned}$$

and finally as:

$$\begin{aligned}
m_{11} &= \cos\theta_1 \cos(\theta_4 + \theta_3 + \theta_2) \\
m_{12} &= -\cos\theta_1 \sin(\theta_4 + \theta_3 + \theta_2) \\
m_{13} &= \sin\theta_1 \\
m_{14} &= \cos\theta_1 [a_4 \cos(\theta_4 + \theta_3 + \theta_2) + a_3 \cos(\theta_3 + \theta_2) + a_2 \cos\theta_2 + a_1] \\
m_{21} &= \sin\theta_1 \cos(\theta_4 + \theta_3 + \theta_2) \\
m_{22} &= -\sin\theta_1 \sin(\theta_4 + \theta_3 + \theta_2) \\
m_{23} &= -\cos\theta_1 \\
m_{24} &= \sin\theta_1 [a_4 \cos(\theta_4 + \theta_3 + \theta_2) + a_3 \cos(\theta_3 + \theta_2) + a_2 \cos\theta_2 + a_1] \\
m_{31} &= \sin(\theta_4 + \theta_3 + \theta_2) \\
m_{32} &= \cos(\theta_4 + \theta_3 + \theta_2) \\
m_{33} &= 0 \\
m_{34} &= a_4 \sin(\theta_4 + \theta_3 + \theta_2) + a_3 \sin(\theta_3 + \theta_2) + a_2 \sin\theta_2 + d_1 \\
m_{41} &= 0 \\
m_{42} &= 0 \\
m_{43} &= 0 \\
m_{44} &= 1
\end{aligned}$$

It is therefore possible to write the following system of four equations and four unknowns:

$$\begin{cases} \cos\theta_1 [a_4 \cos(\theta_4 + \theta_3 + \theta_2) + a_3 \cos(\theta_3 + \theta_2) + a_2 \cos\theta_2 + a_1] = x \\ \sin\theta_1 [a_4 \cos(\theta_4 + \theta_3 + \theta_2) + a_3 \cos(\theta_3 + \theta_2) + a_2 \cos\theta_2 + a_1] = y \\ a_4 \sin(\theta_4 + \theta_3 + \theta_2) + a_3 \sin(\theta_3 + \theta_2) + a_2 \sin\theta_2 + d_1 = z \\ \frac{\sin(\theta_4 + \theta_3 + \theta_2)}{\cos(\theta_4 + \theta_3 + \theta_2)} = \frac{r_{31}}{r_{32}} \end{cases}$$

Dividing the first equation and the second one, it is possible to obtain a simple equation to evaluate θ_1 :

$$\operatorname{tg}\theta_1 = \frac{x}{y}$$

Since from the fourth equation $\cos(\theta_4 + \theta_3 + \theta_2) = r_{32}$, both the first and the second equation can be rewritten as:

$$a_3 \cos(\theta_3 + \theta_2) + a_2 \cos\theta_2 = L_2$$

where

$$L_2 = \frac{x}{\cos\theta_1} - a_4 r_{32} - a_1$$

or

$$L_2 = \frac{y}{\sin\theta_1} - a_4 r_{32} - a_1$$

While, since from the fourth equation $\sin(\theta_4 + \theta_3 + \theta_2) = r_{31}$, the third equation can be written as:

$$a_3 \sin(\theta_3 + \theta_2) + a_2 \sin\theta_2 = L_1$$

where

$$L_1 = z - a_4 r_{31} - a_1$$

At this point it is possible to consider the two equations:

$$\begin{aligned} a_3 \sin(\theta_3 + \theta_2) + a_2 \sin\theta_2 &= L_1 \\ a_3 \cos(\theta_3 + \theta_2) + a_2 \cos\theta_2 &= L_2 \end{aligned}$$

that can be squared and summed together to obtain:

$$\cos\theta_3 = \frac{L_1^2 + L_2^2 - a_3^2 - a_2^2}{2a_2a_3}$$

The calculation of θ_3 gives two possible results, and each one of them can be used to compute θ_2 in the following:

$$\begin{aligned} a_3 \sin(\theta_3 + \theta_2) + a_2 \sin\theta_2 &= L_1 \\ (a_3 \cos\theta_3 + a_2) \sin\theta_2 + (a_3 \sin\theta_3) \cos\theta_2 &= L_1 \end{aligned}$$

If defining:

$$\begin{aligned} a &= a_3 \sin\theta_3 \\ b &= a_3 \cos\theta_3 + a_2 \\ c &= L_1 \end{aligned}$$

it is possible to obtain two values of θ_2 for each value of θ_3 :

$$\begin{aligned} \theta_{2_1} &= \operatorname{atg}(c/\sqrt{a^2 + b^2 - c^2}) - \operatorname{atg}(a/b) \\ \theta_{2_2} &= \operatorname{atg}(-c/\sqrt{a^2 + b^2 - c^2}) - \operatorname{atg}(a/b) \end{aligned}$$

The four θ_2, θ_3 combinations can be evaluated both by taking into account the physical constraints for the two variables and by substituting the couples of values back in one of the equations:

$$\begin{aligned} a_3 \sin(\theta_3 + \theta_2) + a_2 \sin\theta_2 &= L_1 \\ a_3 \cos(\theta_3 + \theta_2) + a_2 \cos\theta_2 &= L_2 \end{aligned}$$

and discarding the couples that do not satisfy the equalities.

Then, for each valid θ_2, θ_3 couple, it is possible to compute θ_4 from:

$$\begin{aligned}\sin(\theta_4 + \theta_3 + \theta_2) &= r_{31} \\ (\cos\theta_2\cos\theta_3 - \sin\theta_2\sin\theta_3)\sin\theta_4 + (\sin\theta_2\cos\theta_3 + \sin\theta_3\cos\theta_2)\sin\theta_4 &= r_{31}\end{aligned}$$

where defining:

$$\begin{aligned}a &= \sin\theta_2\cos\theta_3 + \sin\theta_3\cos\theta_2 \\ b &= \cos\theta_2\cos\theta_3 - \sin\theta_2\sin\theta_3 \\ c &= r_{31}\end{aligned}$$

it is possible to obtain two values of θ_4 for each θ_2, θ_3 couple:

$$\begin{aligned}\theta_{4_1} &= \text{atg}(c/\sqrt{a^2 + b^2 - c^2}) - \text{atg}(a/b) \\ \theta_{4_2} &= \text{atg}(-c/\sqrt{a^2 + b^2 - c^2}) - \text{atg}(a/b)\end{aligned}$$

The feasible triplets can be evaluated both by taking into account the physical constraints for the three variables and by substituting the triplets of values back in one of the equations:

$$\begin{aligned}\sin(\theta_4 + \theta_3 + \theta_2) &= r_{31} \\ \cos(\theta_4 + \theta_3 + \theta_2) &= r_{32}\end{aligned}$$

and discarding the sets that do not satisfy the equalities.

With this closed form solution it is possible to obtain one or more feasible joint configurations for a given position and orientation of the fourth reference frame with respect to the fixed global one. It is important to notice that, with just four degrees of freedom, this section of the arm is not dexterous in a tridimensional space, in the sense that it can bring the end link to a given position in space, but, given the geometrical characteristics of the arm itself, the orientation of the end frame is not completely free. Moreover, the model incorporates knowledge of the physical limitations of servo motors, hence limiting the range in which the angular joint parameters are chosen. Therefore the algorithm is not guaranteed to find a joint configuration for *any* given position and orientation; however it can find one or more joint configurations for *any feasible* position-orientation.

Chapter 3

Analysis and Testing

3.1 Analysis

3.1.1 MDO

Designing an engineering system is a complex task. In particular, the design of aerospace vehicles involves many interacting disciplines, often governed by a very large set of coupled equations. This multidisciplinary nature of the design process leads both to increasing difficulty in handling and solving such problems but also to advances that can be achieved only exploiting the interaction of two or more disciplines.

Traditionally, the engineer has always had to face the challenge of satisfying requirements and constraints while maximizing performance: many problems arise when trying to solve such a problem. Mainly, they can be summarized as:

- Increasing system complexity and need to break the problem up in smaller domains
- Inadequacy of conventional optimization techniques in complex problems
- Discrepancy between design knowledge and design freedom
- Subjectivity in trade-off decisions

The design and analysis of multifunctional systems, being intrinsically multidisciplinary, presents the same issues. Let's spend a few words to describe these problems and give a rationale for the use of Multidisciplinary Design Optimization techniques.

In the pioneering phase of aeronautics, the complexity of flying hardware was relatively low and allowed for a single person to face the whole project autonomously. Later on, with increasing system complexity, the organization of work changed: different actors were made responsible for different subsystems. Each of them had to handle just a fraction of the system equations, now partitioned into subsets.

However, partitioning disciplines often meant to neglect their couplings and therefore to lessen the accuracy of analysis. Moreover, each discipline expert strove

toward the best possible design, but did so in terms of the variables and performances of his own discipline. With this method they caused, albeit unknowingly, detriment to the overall system performance. In other words, separation of different engineering domains and mono-disciplinary optimization did not permit to explore and exploit all design potentialities.

A first mitigation to this “segregated design approach” where everybody is concerned just with its own domain and does not care of affecting other disciplines, came in the 90s with the Concurrent Engineering (CE) approach.

Parametric analysis and optimization techniques have been part of engineering activities for the past fifty years¹. The first optimization tools were merely graphical methods: they were easy to read, use and understand, moreover they had the advantage of showing in a single glance the full behavior of the overall function. Unfortunately, they were not able to catch and show relations in more than three variables.

Later on, with progress in numerical optimization, the same principle was applied to problems in more than 3 dimensions. In fact, even if we cannot visualize it, any design can be defined by a vector in multidimensional space where each design variable represents a different dimension. An MDO procedure can therefore “describe” the objective function in n dimensions and locate its optimum. Unfortunately, primitive MDO techniques were not able to “explain” or “show” the path that led to the optimal solution and consequently inspired too little trust in designers and decision makers.

Conventionally, the development of a product goes through several steps: from conceptual to preliminary to detailed design, followed by manufacturing and production. During this hierarchical sequence of steps that defines the design process, the engineer gains knowledge but loses freedom to act on that knowledge².

Ideally, in order to improve the quality of design, more knowledge should be brought forward to the conceptual and preliminary design phases. In the meanwhile, more design freedom should be retained later into the process, in order to act on the new knowledge gained by analysis, experimentation, and human reasoning (see Figure 3.1). If more design freedom and more information about the product are available in the early stages of design, the engineer should be able to conceive, evaluate and compare a greater number of solutions and/or shorten the design process.

Another interesting aspect is that, when a designer takes a decision, he is expressing a preference that is not only justified by objective engineering or management motivations, but also highly influenced by personal experience, beliefs, intuitions. On the creative side of invention and design, this characteristic is a useful potentiality, but during the evaluation and choice among competing, discretely different concepts, there is the need for each concept to be optimized to reveal its full potential. In this second case, an hasty, biased decision based on incomplete information can be detrimental.

¹The first optimization technique, which is known as steepest descent, goes back to Gauss. Historically, the first term to be introduced was linear programming, which was invented by George Dantzig in the 1940s.

²It was demonstrated mathematically that this natural evolution may lead to suboptimal designs [45]

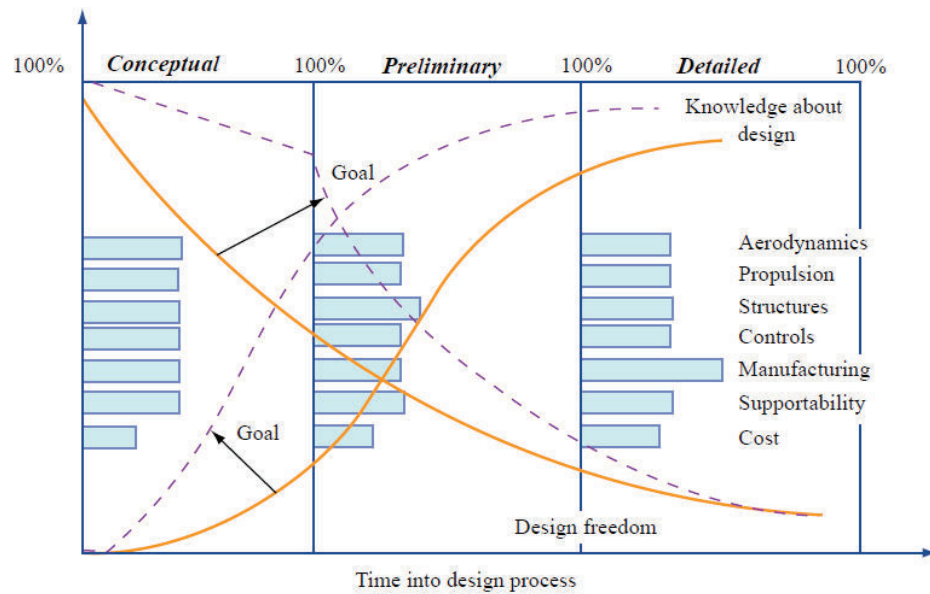


Figure 3.1: Design knowledge and freedom: goal of MDO is to gain knowledge earlier and retain freedom longer (property of MIT OCW - ESD.77 / 16.888).

A possible solution to all these problems lies in a better integration of multi- and inter-disciplinary design, analysis, and optimization.

Multidisciplinary Design Optimization (MDO) can be described as a new methodology for quantitative support of the design process. This methodology calls for an environment in which the engineer and computer can interact exploiting the best resources of each. In particular, the engineer can be in charge of the creative part of the process while the computer can support his decision making with a comprehensive, and quickly generated, numerical information presented in an easy-to-interpret format.

As mentioned in the well known AIAA MDO White Paper on Current State of the Art ([46]) and summarized in the MIT lectures on Multidisciplinary System Design optimization (Olivier de Weck, Karen Willcox), system level MDO has no single definition. It can be described as:

- A methodology for the design of complex engineering systems and subsystems that coherently exploits the synergism of mutually interacting phenomena
- Optimal design of complex engineering systems which requires analysis that accounts for interactions amongst the disciplines
- “How to decide what to change, and to what extent to change it, when everything influences everything else”

It can be noticed that, when optimization is at system level, the process must also include the broader set of concurrent engineering (CE) disciplines (manufacturing, supportability, cost. . .). This is a Gordian node of MSDO, because the development of realistic, reliable and easy to use mathematical models for these CE disciplines is very difficult.

Of course, it is worth stressing the concept that human element is a key component in any successful system design methodology. Therefore, it is wrong to think of MDO as an “automated” or “pushbutton” design methodology: it will always be an interactive process where the engineer retains the role of meta-designer and decision maker.

There is one final remark. In the previous discussion, MDO has been depicted as a powerful tool for the selection of an optimal design. However, it is important to mention that, given the fact that modern engineering problems are really difficult to solve, often MDO is used not to find the truly optimal design, but rather to find an improved design, or even a feasible design.

A more theoretical and detailed description of MDO techniques will be given in the remainder of this section.

Fundamentals of MDO

First of all, it is important to define a number of terms.

Design variables are parameters that the engineer can modify in order to change the product he is designing. There are several different typologies of design variables, but often the term is used as a synonym for independent design variable or design parameter.

An independent design variable or design parameter is the actual quantity that the engineer can handle: geometry, materials, configuration. . .

On the other hand there are dependent design variables (also called attributes or characteristics), that is to say, quantities that the designer cannot change directly, but which he can affect and manipulate via the design parameters. These variables determine the “value” of the design.

Independent and dependent variables are the main quantities to work with, nonetheless it is also possible to define:

- State variables: artificial intermediates between dependent and independent design variables
- Operating variables, environmental variables or external variables: parameters that can be changed after the product has actually been built and can affect its performance when used

Any design can be defined by a vector \mathbf{x} in multidimensional space, where each of the n design variables x_i represents a different dimension. The x_i can be real, integer, boolean, binary. . . Independent design variables are subject to constraints and boundaries.

Boundaries are explicit limits on the variability of design parameters. Constraints act as boundaries of the design space and typically occur due to finiteness of resources or technological limitations of some design variables.

- Bounds:

$$x_{i,LB} \leq x_i \leq x_{i,UB} \quad i = 1, 2, \dots, n$$

- Inequality constraints:

$$g_j(x) \leq 0 \quad j = 1, 2 \dots m_1$$

- Equality constraints:

$$h_k(x) = 0 \quad k = 1, 2 \dots m_2$$

It is important to notice that the final optimum design is influenced by boundaries and constraints. Often, but not always, optimal designs lie at the intersection of several active constraints.

The value of independent variables influences that of dependent variables. In turn, dependent variables are used to compute the objective function value. The objective function can be built taking into account measures of system behavior (performance), resource utilization (time, money, fuel...) or risk (stability margins...). Generally speaking, the objective can be a vector \mathbf{J} of z system responses or characteristics that the designer is trying to maximize or minimize: often the objective is just a scalar function, but for real systems suitable for multi-objective optimization it is necessary to deal with vector notation.

Last, but not least, there is the definition of parameters (in this case, fixed parameters, not design parameters). Parameters p are quantities that affect the objective \mathbf{J} , but are considered fixed, i.e. they cannot be changed by the designers. Sometimes parameters p can be turned into design variables x_i to enlarge the design space. Sometimes parameters p are former design variables that were fixed at some value because they were found not to affect any of the objectives J_i or because their optimal level was predetermined.

Now that notation is clear, it is possible to describe the MDO process. The design vector \mathbf{x} contains n variables that form the design space; during design space exploration or optimization entries of \mathbf{x} are changed in some rational fashion to achieve a desired effect. In other words, MDO mathematically traces a path in the design space from some initial design \mathbf{x}_0 towards improved designs (with respect to the objective \mathbf{J}).

In formal notation, this can be written as:

$$\begin{aligned} & \text{minimize } \mathbf{J}(\mathbf{x}, \mathbf{p}) \\ & \text{subject to} \\ & \mathbf{g}(\mathbf{x}, \mathbf{p}) \leq 0 \\ & \mathbf{h}(\mathbf{x}, \mathbf{p}) = 0 \\ & x_{i, LB} \leq x_i \leq x_{i, UB} \quad i = 1, 2 \dots n \end{aligned}$$

where

$$\begin{aligned} \mathbf{J} &= [J_1 \ J_2 \ \dots \ J_z]^T \\ \mathbf{x} &= [x_1 \ x_2 \ \dots \ x_n]^T \\ \mathbf{g} &= [g_1 \ g_2 \ \dots \ g_{m_1}]^T \\ \mathbf{h} &= [h_1 \ h_2 \ \dots \ h_{m_2}]^T \end{aligned}$$

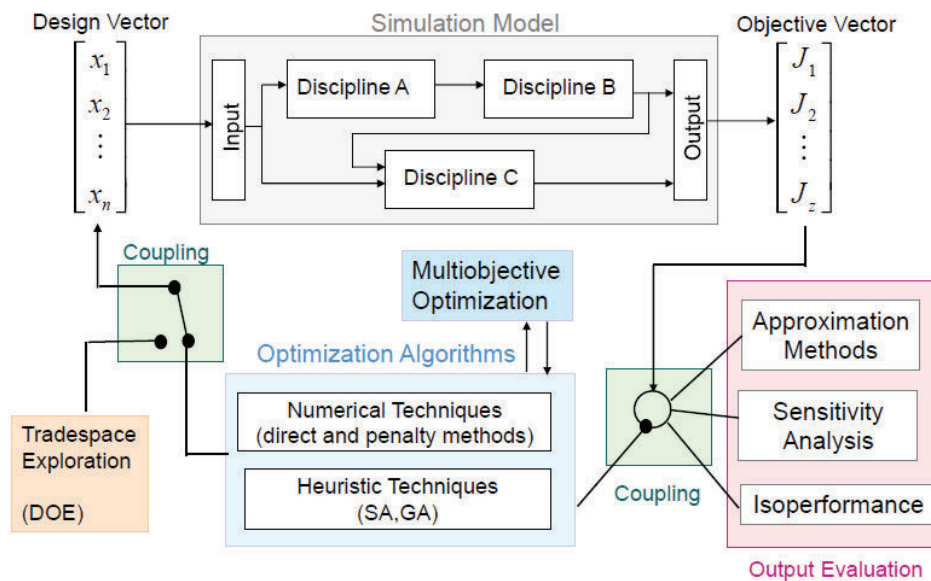


Figure 3.2: Schematic description of the MSDO process (property of MIT OCW - ESD.77 / 16.888).

Figure 3.2 can help understand the complexity of a MDO process. Starting from top left corner, we can see the design variables vector, that is what the engineer can directly change. Given the complexity of the problem and the computational effort needed to evaluate consequences of changes in these quantities, design variables are usually changed following a strategy. This strategy can be both a rational method to explore the design space (box in the bottom left corner) or an optimization algorithm to steer the design toward an improved solution (boxes in the bottom center of Figure 3.2). Independent design variables are fed into a simulation model (top center of Figure 3.2), used to assess the design value: it can check constraints and bounds, compute dependent design variables and quantify objective functions. The output is the objective vector (tip right corner). Based on this vector, the optimization algorithm can adjust the design parameters to improve the design value. As an alternative, the objective vector can be analyzed in order to catch the relationship between its value and input variables, fixed parameters... This kind of analysis is a useful post-processing activity that can evaluate the reliability of the solution or discover hidden aspects of the problem (box in the bottom right corner of the picture).

Therefore, the MDO process can be described as follows:

1. Definition of overall system requirements: this step is necessary to establish and evaluate the objective function.
2. Definition of design vector \mathbf{x} , objective \mathbf{J} and constraints: this step benefits from conceptual design activities, as MDO best suits refinements of detailed design.
3. System decomposition into modules: this steps is mainly concerned with the complex task of isolation of different disciplines and definition of interfaces among them.

4. Modeling of physics via governing equations at the module level (module execution in isolation): this step must turn out in the creation and validation of a model of adequate degree of complexity for each distinct discipline/module.
5. Model integration into an overall system simulation: this step is often solved via dedicated software that handles models execution, interfaces, data exchange and the overall simulation.
6. Benchmarking of model with respect to a known system from past experience, if available: this step is fundamental in order to verify congruence and obtain meaningful solutions.
7. Design space exploration to find sensitive and important design variables x_i : this step is essential in exploring the domain of design variables and discovering their effect on design objectives.
8. Formal optimization to find $\min \mathbf{J}(\mathbf{x})$: this step is the actual process of adjusting \mathbf{x} in order to find the optimal solution.
9. Post-optimality analysis: this is a post-processing step that should include uncertainty, explore sensitivity and tradeoffs.

Often the optimization process leads to mathematically valid but physically unreasonable solutions or to solutions that are not feasible based on system constraints. In these cases there is the need to add, remove or modify constraints and/or design variables in order to obtain an appropriate response.

Let's go back for a moment to Figure 3.2. Focusing on the inner optimization loop, it is clear that the MDO process needs two distinct components:

- The optimization algorithm, which decides how to move through the design space.
- The simulation model, which evaluates designs chosen by the optimizer (sometimes, disciplinary simulation models can be used in an optimization framework, but often they are not appropriate).

In order for the MDO technique to be effective, the multidisciplinary description of the system must be reliable and sufficiently detailed. Nowadays, many disciplines can benefit from models of incredible completeness but these models, as already mentioned before, are not always the best choice in MDO environments. This is because specific softwares capable of simulating the innermost detail of a certain phenomenon are often computationally expensive and inefficient. A multidisciplinary analysis is a process that requires substantial computational effort on its own, this is the reason why simpler discipline models can be the key to an MDO environment with solution times compatible or, better, competitive with industrial project schedule.

Two Figures can help understand this problem.

The first one depicts the fidelity level of discipline models versus the level at which MDO should be applied. It is easy to understand that an increasing model

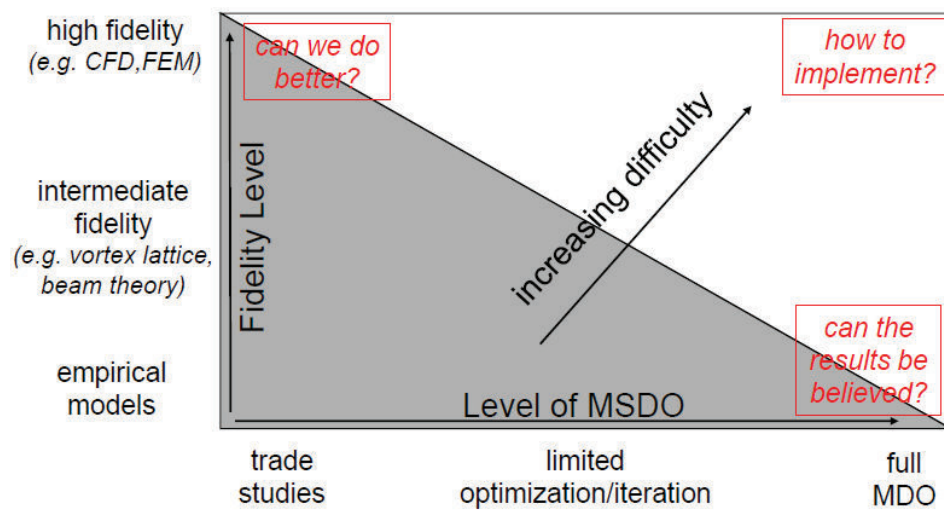


Figure 3.3: MSDO process: fidelity versus analysis level (property of MIT OCW - ESD.77 / 16.888).

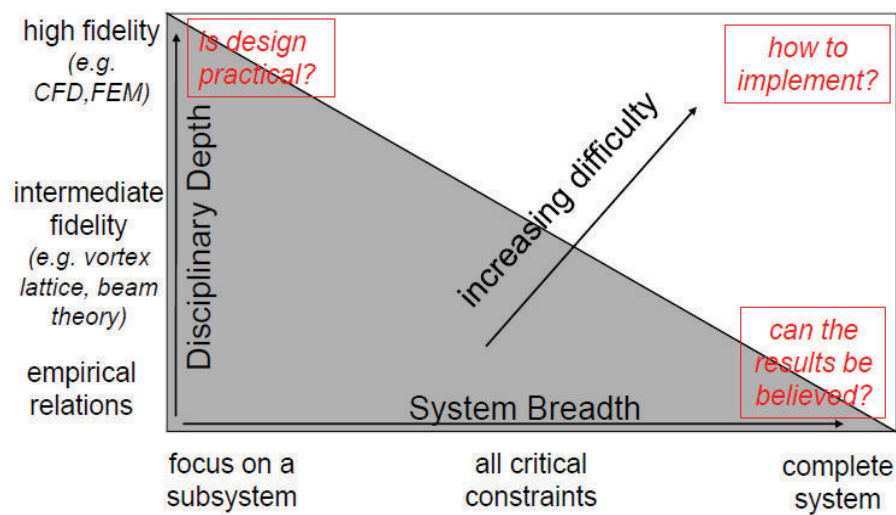


Figure 3.4: MSDO process: analysis depth versus system breadth (property of MIT OCW - ESD.77 / 16.888).

fidelity hinders the usage of MDO at full system level. Besides this, one can also notice that simple empirical models, while computationally allowing full MDO, do not grant credibility to its results.

The second picture shows the discipline depth (strictly related to model fidelity) versus the system breadth. Its interpretation is very similar to the previous Figure: great discipline depth (detailed models) is suitable for optimization at subsystem level, while little discipline depth (simple models) can enable system level MDO, but with not well-founded results.

In both cases, the difficulty of applying high fidelity models to full system level MDO is highlighted.

With these preliminary remarks it is clear that the simulation model has a central role in the determination of MDO tool performances. Let's see some guidelines on how to build a correct simulation model.

In MDO field, a model is a mathematical object that has the ability to predict the behavior of a real system under a set of defined operating conditions and simplifying assumptions. A mathematical model is, basically speaking, a set of equations and boundary conditions (respectively describing system physics and environment). There are analytical mathematical models and empirical mathematical models. The former ones are based on differential equations or finite differences approaches, the latter ones are based on laboratory data interpolation.

In any case, a model is just an abstract description of the system and therefore it should include only those aspects that are pertinent to the scope of the study. In other words a model should include only components and interactions that are relevant to the aim of the simulation.

The overall simulation architecture should be composed of modules. In this context, a module is a finite group of tightly coupled mathematical relationships that are under the responsibility of a particular individual or organization³, and where some variables represent independent inputs while others are dependent outputs. Therefore a module performs a compact set of calculations, contains a single entry point and exit point and should be testable in isolation. For this reason, modules separation is mainly based on disciplinary separation and related analysis software: a module gathers governing equations with high degree of coupling. Weaker couplings are represented via interfaces with other modules.

The process of identifying and separating modules is called decomposition. Once decomposition has been carried out, the set of black boxes should be organized in an ordered arrangement, based on input/output relationships. At the end of this process, the simulation architecture should present high internal coupling within the module, because a single module should correspond to a single primary function, low coupling between modules, because the number of variables that are exchanged should be minimized, and minimization of feedback loops, because data flow should be processed sequentially from input to output.

A useful tool in this architecture organization task is the N^2 diagram, an N by N matrix where each module is placed along the diagonal while inputs and outputs are represented, respectively, along columns and rows. Such a visual representation of information flow helps identifying critical modules, organizing execution sequence,

³A module can frequently appear as a "black box" to other individuals or organizations.

grouping correlated modules and minimizing feedback loops.

Thanks to this structure, coding and testing of modules can be done in parallel, they can be integrated in the correct sequence and properly validated before release of simulation model.

From a practical point of view, the simulation architecture can be built in several different ways:

- Several different modules implemented in a single application on the same computer.
- Several different modules implemented on the same computer but under different applications.
- Several different modules implemented under different applications on different computers connected in LAN or WAN environment.

Modular analysis and optimization can be performed as a sequential process or in a distributed fashion. In this second case, there are two possible approaches:

- Distributed analysis.
- Distributed design.

In distributed analysis approach, disciplinary models provide analysis and evaluate constraints/objectives while optimization is controlled by some overseeing code or database: this supervising actor keeps track of the values of the design variables and objective and changes the values of the design variables according to the optimization algorithm.

In distributed design approach, disciplinary models are provided with design tasks, and optimization takes place at subsystem and system levels. Under this category, there are two main techniques: Concurrent Subspace Optimization (CSSO) and Collaborative Optimization (CO).

In CSSO, the design problem is divided into several discipline-related subspaces, where each subspace shares the responsibility for satisfying constraints while trying to reduce a global objective.

In CO disciplinary teams satisfy local constraints while trying to match target values specified by a system coordinator. In this technique, optimization takes place at two different levels: a system-level optimization provides a set of targets, chosen to optimize the system-level objective function, while a subsystem optimization finds a design that minimizes the difference between current states and the targets. With this method disciplinary-level design freedom is preserved because the subsystem optimizer can modify local variables to achieve the best design for which the set of local variables and computed results most nearly matches the system targets and local constraints.

Traditional design process often follows an incremental path:

- Generate ideas.
- Assess value of ideas.

- Select best idea.
- Incrementally improve best idea.
- Implement improved idea.

This traditional approach made of many small steps limits the exploration of the design space; moreover it is also problematic when dealing with novel domains.

In order to overcome these drawbacks, engineers can capitalize on Design Space Exploration (DSE) techniques. In industrial settings, the primary goal of experimental techniques is usually to extract the maximum amount of unbiased information regarding the factors affecting a certain objective from as few (costly) observations as possible. In particular, design space exploration is the process of analyzing several alternative points in the design space and its main purposes are:

- Identify key drivers among potential design variables, identify appropriate design variable ranges, discover influences of different variables on design value, understand relevant features underlying the design space⁴.
- Discover feasibility regions (even the “odd shaped” or “odd placed” ones), identify new interesting points.
- Identify achievable objective function values.
- Support generalization⁵ and prediction⁶ for different designs within the same class of requirements.
- Help tradeoff among functionally equivalent alternatives.

Once the design space is clearly understood, the engineer can focus on a specific subset of it and optimize a single design point.

Of course, the preliminary design space exploration for large, interdisciplinary engineering problems is a difficult and time-consuming task.

There are several techniques to perform a design space exploration. One of these techniques is the Design of Experiments (DoE), a collection of statistical techniques providing a systematic way to sample the design space. In the DoE theory, design variables are named “factors” whilst their possible values are called “levels”. There are two different kinds of factors: control factors (i.e. variables the designer can control) and noise factors (i.e. variables the designer cannot control).

The combination of different factors with their respective levels is given as an input for an experiment. The experiment often consists in running a simulation or analysis code. The outcomes are called observations and represent the objective value produced by that particular set of inputs.

Schematically, this can be described with a matrix where columns represent different factors and rows contain different levels of each factor. Therefore, with this

⁴DSE increases synthetic knowledge (“What are the conditions for a desirable outcome?”).

⁵DSE increases descriptive knowledge (“Why things are like they are?”).

⁶DSE increases predictive knowledge (“What is the outcome for a given condition?”).

Table 3.1: Tabular structure of experiments in DoE.

Experiment number	Factor A	Factor B	Observation
1	A_1	B_1	η_1
2	A_1	B_2	η_2
3	A_2	B_1	η_3
4	A_2	B_2	η_4

structure, each row represents a different experiment. The first and last columns report, respectively, the ordinal number and the outcome of each experiment.

In a hypothetical experiment with two factors with two levels each, the experiment matrix would be similar to that in Table 3.1.

In the following paragraphs, different approaches to DoE will be presented and briefly described.

Full factorial DoE is the complete examination of all factors at each level: every possible combination is enumerated. If we designate as f the number of factors and l the number of levels for each factor, the quantity of experiments e to be performed is $e = l^f$. It is simple to understand that this method is computationally very expensive. Of course, a factorial design allow the effect of each and every factor to be tested and estimated independently with the interactions also assessed.

Designs having all input factors set at two levels each are called 2^k full factorial designs. The two levels are called high or $+1$ and low or -1 . As a drawback, one must notice that a two-level experiment can not fit quadratic effects. The remedy for dealing with significant curvature in two-level factorial design is to add more points: for further detail see the paragraph on Central Composite Design.

In order to reduce the number of experiments to be evaluated, a simple solution could be to select a subset of the full-factorial design space and perform analysis only on those cases. This is called the fractional factorial DoE approach. Of course, two main open points arise with such a method: how many cases should be considered, and how should those cases be selected? There is no complete answer and often the guideline is to carefully balance coverage needs versus computational costs. As a rule of thumb, it can be useful to examine a large number of factors at a reduced number of levels instead of analyzing a great number of levels for a limited number of factors.

Plackett-Burman DoE is derived from full-factorial matrices using the assumption that all interactions are insignificant relative to main factor effects. It is based on the extraction of screening experiments matrices (two level fractional factorials) from full-factorial experiments matrices. Using a limited number of experiments, this DoE minimizes the variance of the estimates of the dependencies. Main factor effects in Plackett-Burman designs replace all interactions of the full factorial matrix. However, this does not mean that information about interactions cannot be determined from Plackett-Burman designs.

Screening experiments identify the “power factors” in a process: these factors are the process variables that most significantly affect the process outputs, or responses. Regardless of whether the experimental objective is to maximize the response, minimize the response, or hit a target, the power factors are the critical process or product variables used to adjust and control the output.

The Plackett-Burman design of experiments is mainly used in matrix form: columns represent factors and rows contain process runs. Factors go across the top, labeled f_1, f_2, \dots, f_n , while the number of runs will go down columns. Cells containing the minus sign (-) will have lower limits per variable and cells containing the plus sign (+) will have upper limits per variable. On the last run there are all (-) signs. Once the experiment is run and the samples measured, it is time for the post-processing of results: data from the experiment are used to calculate the effects and to determine the statistical significance of those effects.

In parameter study DoE approach, a baseline level for each factor is identified. Then, the method consists in evaluating all possible levels for a single variable, while other variables are fixed at baseline level. This is done separately for every factor. Assuming that all factors f have the same number of levels l , the amount of experiments to be carried out can be computed as: $e = 1 + f(l - 1)$. This procedure can identify the best value for each factor, but cannot spot interactions between variables. On the other hand it is computationally cheap. It is highly unlikely that the “best result” (all factors at their optimum level) will be part of the experiments set, however it can be “extrapolated” from the analysis of each factor’s behavior (interactions are omitted, though).

The one at a time DoE method requires the definition of a baseline value for each variable; moreover, factors must be ordered from first to last one. Then, the first factor is evaluated at each value: this allows to select the optimal level. From this point on, this factor remains fixed at its optimal level and the second one is evaluated at each value (while all other factors are still at baseline value). This time the process identifies the optimal level for the second factor and blocks it. The procedure goes on until all factors have been evaluated; the total number of experiments is $1 + f(n - 1)$. Unfortunately, results are highly dependent upon the order of factors. The “best result” (all factors at their optimum level) will be part of the experiments set, but it will miss some of the interactions between variables.

In the latin hypercubes DoE method, the design space is partitioned into l divisions for each factor f . Then, divisions are combined randomly in order to generate l design points while using each level of a factor only once. Unfortunately, due to the random nature of choice, this method’s results are not repeatable and they can show poor coverage. As a response to this second drawback, an improvement of this approach, called optimal latin hypercube DoE, has been proposed: its main characteristic is that it is space-filling, i.e. there is an even distribution of samples in the design space. An optimal latin hypercube strategy can exploit different techniques in order to generate an appropriate set of design points: random search, genetic algorithm, enhanced stochastic evolutionary algorithm. . .

The orthogonal arrays DoE approach, like the fractional factorial one, selects a subset of the full factorial design space. The criterion that leads the selection is

orthogonality of samples. Referring to matrix notation, in order to obtain orthogonality, the following must hold: for any pair of columns, all combinations of factor levels occur and they occur an equal number of times. This is called the “balancing property” and, in general, it is sufficient for orthogonality.

The central composite design falls under the classical quadratic design category, where a fractional plan is used to fit a second order equation. They start with a factorial or a fractional factorial design (with center points) and then “star points” or “axial points” are added to estimate curvature. These points are located along the axes of the factor space. If constructed properly, the CCD provides a solid foundation for generating a response surface map.

For maximum efficiency, the axial or star points should be located a specific distance outside the original factor range. The ideal location can be found through research or calculated by software, but it will be very close to the square root of the number of factors. the presence of additional centerpoints provides a link between the blocks⁷ and add more power to the estimation of second order effects needed to characterize curvature. In this configuration, the CCD contains five levels of each factor: low axial, low factorial, center, high factorial and high axial; therefore it generates enough information to fit a second order polynomial.

Once the experiments have been performed, the results can be used to compute effects: the effect of a factor is the change in the response as the level of the factor is changed. Often, the sign of the effect is considered: e.g. in a 2-level experiment, the effect is determined for a change from a low (minus) level to a high (plus) level.

There are basically two types of effects:

- Main effects.
- Interaction effects.

The main effect of a factor is the effect of that factor on the output averaged across the levels of other factors.

The interaction effect between factor f_1 and factor f_2 is defined as half the difference between the average f_1 effect with f_2 at high level and the average f_1 effect with f_2 at low level.

In building a DoE, the goal is always to allow the experimenter to evaluate in an unbiased (or least biased) way, the consequences of changing the settings of a particular factor, that is, regardless of how other factors were set. In more technical terms, it is an attempt to generate designs where main effects are unconfounded among themselves, and in some cases, even unconfounded with the interaction of factors.

The Multidisciplinary Design Optimization techniques pave the way for a new design procedure that brings both advantages and disadvantages. Among the possible benefits, one can list:

- It enables reduction in design time.

⁷Block 1 is made of the “old” factorial points, while block 2 is composed of the “new” star points.

- It allows for a reduction in design costs and, consequently, lifecycle costs.
- It is a systematic, logical design procedure.
- It is not biased by intuition or experience.
- It can handle a wide variety of design variables and constraints.

In contrast, the same procedure can produce drawbacks, especially when it is implemented for the very first time:

- It requires huge computational resources, especially when dealing with complex systems (computational time grows rapidly and numerical problems increase with number of design variables).
- It requires an initial effort to model disciplines and build interfaces.
- Its results are reliable only after a careful post-processing analysis.

Furthermore, there are a number of open points that still have to be solved:

- System description: there is the need for some kind of database to store design variables, constraints, objectives. . .
- Balance between fidelity and expense of disciplinary models.
- Fundamental choices to handle problem complexity: design variables, constraints, and model interfaces must be managed carefully.
- Ease of use: communication with the final user is often very difficult because the interface is frequently very unfriendly and it can be laborious to change problem parameters.
- Lack of flexibility: a general purpose MDO tool is not easy to build because it is common for such a tool to become very specialized and only valid for one particular problem.

3.1.2 Optimization methods

This section reviews search algorithms that are useful when no analytical solution is available for an optimization problem. It should be noted that, when there is availability of an analytical solution, a search algorithms should not be used, because the analytical solution gives a global optima whereas there is always a possibility that the numerical solution may get stuck at local optima.

Generally speaking, optimization methods are algorithms able to approach a proximity of a function maximum⁸ while allowed to evaluate the function in a finite number of points. The search methods therefore approximate the behavior of the function in the neighborhood of measurements.

⁸Since any maximum of $f(x)$ is a minimum of $-f(x)$, we will consider indiscriminately maximization or minimization problems.

In order for an optimization algorithm to perform well, the objective function should or should not have some specific characteristics. Usually, some properties of the objective function are a priori assumed. Without assumptions, no rational algorithms can be suggested. Very often, the approximation is based on the assumed smoothness or sometimes the convexity: various methods assume different types of the approximation.

The maxima (or minima) of a function can either be global or local: in principle, we are most interested in finding the global optimum, but this can be very difficult. Generally, there are no ways to predict the behavior of the function everywhere in the permitted domain: often a problem will have many local optima (perhaps caused by noise) which means that locating the single global optimum can be tricky. This happens as well because most of the methods pilot the search to a local optimum without a guarantee that this optimum is also a global one. Those methods that guarantee the global character of the optimum, require additional assumptions e.g. the convexity.

As mentioned before, the most suitable methods to locate minima depend upon the nature of the function which is considered. There are two broad classes of algorithms:

- Local optimizers that, given a point in a valley of the function, locate the optimal point on the valley.
- Global optimizers that range over a region of parameter space attempting to find the optimum of the most pronounced valley.

If a good estimate of the position of the optimum exists, it is a good idea to only use a local optimizer to improve it and find the optimum choice of parameters. If no such estimate exists, some global method must be used. Often an application will require both local and global methods.

Additionally, a characteristic of special interest is the sharpness of the optima. It is very difficult to find an optimum of a flat function, especially in the presence of numerical errors. On the other hand, a sharp optimum also requires some special methods since the gradient is discontinuous.

Also, the presence of constraints and their character is very important for the choice of the strategy. The constrained problem typically possess the sharp maximum in a corner of the permitted domain while the smooth unconstrained function reaches its maximum in a stationary point.

Besides the classification “local vs. global optimizer” there are many other categories, for example:

- Number of variables: optimization methods are designed for various number of independent controls (dimensionality), these ranges corresponds to one variable, several (two to ten), dozens, hundreds, or tens of thousands.
- Cost of evaluation: it is a significant parameter of the function (it ranges from milliseconds of the computer time to lengthy and costly measurements in natural experiments) and the realistic goals and methods are different accordingly to it.

- Discrete optimization: it is an important special class and corresponds to a large number of controls that take only integer or boolean values, it requires combinatorial methods.
- Calculus of variations: it is another special case which corresponds to infinitely many controls.
- Stochastic optimization: it deals with noise in measurements or uncertainty of some factors in the model, and therefore accounts for possible inaccuracies in data.

Another possible approach is to divide unconstrained versus constrained optimization techniques. Among the techniques to solve the former we have steepest descent, conjugate gradient, quasi-newton, newton... while to solve the latter we have simplex, SLP (Sequential Linear Programming), SQP (Sequential Quadratic Programming), exterior penalty, interior penalty, generalized reduced gradient, feasible directions, mixed integer programming...

In constrained optimization, a feasible design is a design that satisfies all constraints, while an infeasible design is a design that violates one or more constraints. Obviously, the optimum design is the set of design variables that minimizes the objective function while satisfying all constraints. In general, constrained optimization algorithms try to cast the problem as an unconstrained optimization, and then use a conventional gradient or derivative-free technique to solve that problem.

In fact, a constrained minimization problem with m_1 inequality constraints and m_2 equality constraints can be stated as an unconstrained minimization problem by defining the Lagrangian function:

$$L(\mathbf{x}, \lambda) = J(\mathbf{x}) + \sum_{j=1}^{m_1} \lambda_j g_j(\mathbf{x}) + \sum_{k=1}^{m_2} \lambda_{m_1+k} h_k(\mathbf{x})$$

where $L(\mathbf{x}, \lambda)$ is called the lagrangian function, and λ_j is the j -th Lagrange multiplier.

This approach is useful because it can be shown that a stationary point \mathbf{x}^* of $L(\mathbf{x}, \lambda)$ is a stationary point of the original constrained minimization problem.

In particular, if \mathbf{x}^* is optimum, these conditions are satisfied:

- \mathbf{x}^* is feasible.
- $\lambda_j g_j(\mathbf{x}^*) = 0 \quad j = 1 \dots m_1 \text{ and } \lambda_j \geq 0$
- $\nabla J(\mathbf{x}^*) + \sum_{j=1}^{m_1} \lambda_j \nabla g_j(\mathbf{x}^*) + \sum_{k=1}^{m_2} \lambda_{m_1+k} \nabla h_k(\mathbf{x}^*) = 0$, with $\lambda_j \geq 0$ and λ_{m_1+k} unrestricted in sign.

These are called Karush-Kuhn-Tucker (KKT) conditions, and they are necessary and sufficient optimality conditions if the design space is convex. They can be interpreted like this:

- Condition 1: the optimal design satisfies the constraints.

- Condition 2: if a constraint is not precisely satisfied, then the corresponding Lagrange multiplier is zero, i.e. the j -th Lagrange multiplier represents the sensitivity of the objective function to the j -th constraint, and it can be thought of as representing the “tightness” of the constraint: if λ_j is large, then constraint j is important for this solution.
- Condition 3: the gradient of the Lagrangian vanishes at the optimum.

Many optimization algorithms are iterative, i.e. they gradually improve the value of the objective function as:

$$\mathbf{x}^q = \mathbf{x}^{q-1} + \alpha^q \mathbf{S}^q$$

where the initial solution x^0 is given, q is the iteration number, α is the scalar distance, and S is the vector search direction. The search direction S is determined according to some criteria: gradient-based algorithms use gradient information to decide where to move, while gradient-free algorithms use sampling and/or heuristics.

In the world of optimization, there is no general consensus on nomenclature or taxonomy of optimization techniques. The classification and definitions used in the remainder of this section are mainly taken from two sources: Prof. Kumar of the India Institute of Science, Bangalore, and Prof. Willcox of the Massachusetts Institute of Technology. However, the author is aware that in other fields, like in control optimization, there are different conventions.

The taxonomy that will be detailed in the remainder of this section is that of Figure 3.5.

Linear programming (LP) intends to obtain the optimal solution to problems that are perfectly represented by a set of linear equations, thus requires a priori knowledge of the problem.

$$\begin{aligned} \min J(\mathbf{x}) &= \sum_{i=1}^n c_i x_i \\ \sum_{i=1}^n a_{ij} x_i &= b_j & j = 1 \dots m \\ x_i &\geq 0 & i = 1 \dots n \end{aligned}$$

or

$$\begin{aligned} \min J(\mathbf{x}) &= \mathbf{c}^T \mathbf{x} \\ \mathbf{A} \mathbf{x} &= \mathbf{b} \\ x_i &\geq 0 & i = 1 \dots n \end{aligned}$$

To convert inequality constraints to equality constraints, it is possible to use additional design variables called slack variables:

$$\sum_{i=1}^n a_{ji} x_i \leq b_j$$

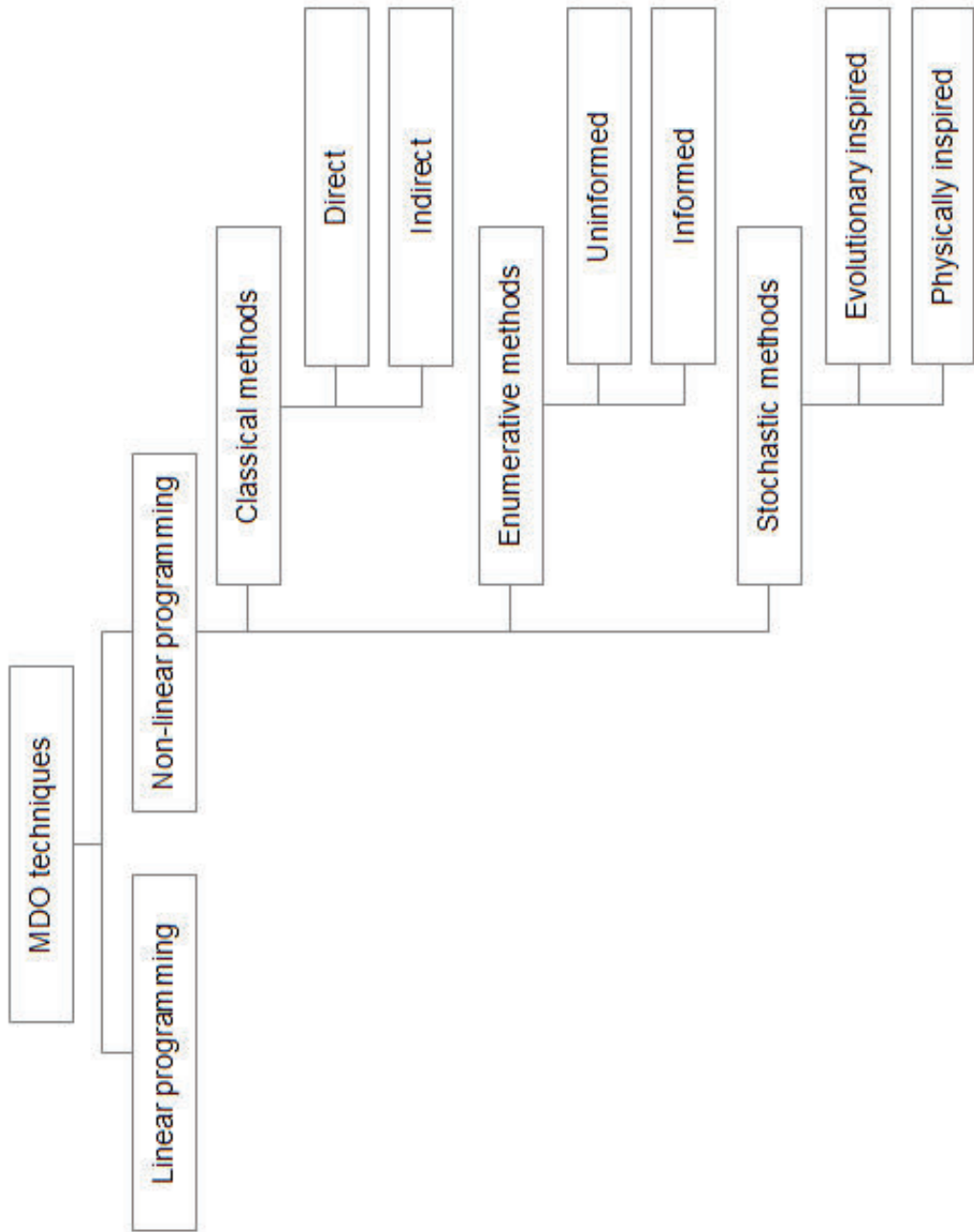


Figure 3.5: A taxonomy of optimization techniques.

becomes

$$\sum_{i=1}^n a_{ji}x_i + x_{n+1} = b_j$$

where $x_{n+1} \geq 0$ is the slack variable. Even a general nonlinear problem can be solved by LP if it is linearized via first order Taylor series: usually linear programming implements the following sequence.

- Pick an initial guess \mathbf{x}^0 .
- Linearize about \mathbf{x}^0 using first-order Taylor series.
- Solve the resulting LP to find $\delta\mathbf{x}$.
- Update the solution $\mathbf{x}^1 = \mathbf{x}^0 + \delta\mathbf{x}$
- Linearize about \mathbf{x}^1 and repeat: $\mathbf{x}^q = \mathbf{x}^{q-1} + \delta\mathbf{x}$, where $\delta\mathbf{x}$ is the solution of an LP (problem linearized about \mathbf{x}^{q-1}).

Unfortunately, there are a number of drawbacks with LP. First of all the linear approximation is only valid close to \mathbf{x}^0 , furthermore there is the need to restrict the size of the update $\delta\mathbf{x}$. For these reasons, LP is not considered to be a good method.

Non-linear programming is intended for problems described by non-linear equations. The methods that go under this class are divided in three large groups: classical, enumerative and stochastic.

- Classical (or calculus based) search uses deterministic approach to find the best solution, which satisfies a set of necessary and sufficient conditions of the optimization problem. Classical techniques include direct and indirect methods.
 - Direct optimization methods depend on the objective function only through ranking a countable set of values. They do not involve derivatives of the function, and since they neither compute nor approximate⁹ derivatives they are also called non-gradient, or derivative-free, or zero-th order methods. They basically try to find extremes by hopping around the search space, moving in a direction related to the local gradient (but not obtained by explicit evaluation of the gradient itself). They can be further divided into pattern search methods¹⁰, simplex methods¹¹, and

⁹E.g. by finite differencing.

¹⁰Pattern search methods are characterized by a series of exploratory moves that consider the behavior of the objective function at a pattern of points, all of which lie on a rational lattice. These exploratory moves consist of a systematic strategy for visiting the points in the lattice in the immediate vicinity of the current iterate.

¹¹Simplex methods are based on the observation that it should take no more than $n + 1$ values of the objective to identify a downhill (or uphill) direction. This makes sense, since $n + 1$ points in the graph of $f(\mathbf{x})$ determine a plane, and $n + 1$ values of $f(\mathbf{x})$ would be needed to estimate the gradient via finite differences. At the same time, $n + 1$ points determine a simplex, a set of $n + 1$ points in \mathbf{R}^n (a triangle in \mathbf{R}^2 , and tetrahedron in $\mathbf{R}^3 \dots$), so it is possible to construct a nondegenerate simplex in \mathbf{R}^n and use the simplex to drive the search, which provides a frugal design for sampling the space and the nice characteristic that reflecting a vertex through the centroid of the opposite face the result is also a simplex. The ultimate goals are either to replace the best vertex or to ascertain that the best vertex is a candidate for a minimizer.

methods with adaptive sets of search directions¹². Some examples are: random search method, grid search method, univariate method, pattern directions, Rosenbrocks method of rotating coordinates, simplex method.

- Indirect optimization methods are also called descent methods. They are based on the derivatives or gradients of the objective function. They work straight on differential models without taking into account a discretization: they search for local extremes by solving usually non-linear sets of equations resulting from setting the gradient of the objective function to zero. They are called “indirect” because they do not allow direct control over the performance index, but they enable reaching the optimum by an error reduction procedure. These methods can provide the exact solution to a given problem with very low computational costs, provided that the initial solution estimate is near enough to the real optimum (otherwise the method is not robust). Indirect methods can be used for solving only unconstrained optimization, while handling of constraints is more problematic than with direct methods. In order to solve constrained optimization problems, a common procedure is the use of a penalty function to convert the constrained optimization problem into an unconstrained optimization

problem. Their application in MDO is extremely limited: they can be used for optimizations in the time domain with analytical mathematical models. Some examples are: steepest descent (Cauchy) method, conjugate gradient (Fletcher-Reeves) method, Newton method, Marquardt method, quasi-Newton method.

- Enumerative search is based on evaluating each possible solution from a given finite set and testing them to find optimal solution. It is easy to implement, but usually require significant computation, thus these techniques cannot scale up and are not suitable for applications with large domain spaces. In the field of artificial intelligence, the enumerative methods are subdivided into two categories: uninformed or blind methods, which search all points in the space in a predefined order (e.g. Mini-Max algorithm) and informed methods, which does more sophisticated search using domain specific knowledge in the form of a cost function or heuristic to reduce cost for search (e.g. Alpha-Beta and A*). The enumerative search methods explore the search space “intelligently” which means evaluating possibilities without investigating every single case. There are many control structures for search algorithms: the depth-first search and breadth-first search are two common search strategies.
- Stochastic search deliberately introduces randomness into the search process. Unlike the previous two search methodologies, evolutionary or stochastic methods include heuristics and an element of randomness (non-determinism) in traversing the search space. The injected randomness may provide the necessary impetus to move away from a local solution when searching for

¹²This family includes Rosenbrock’s and Powell’s methods. These algorithms attempt to accelerate the search by constructing directions designed to use information about the curvature of the objective obtained during the course of the search.

a global optimum. All stochastic methods are based on the creation of a random sequence of solutions (families), and an evolution and recombination of them. Evolution and recombination are usually based on rules dictated by the physical problem and they favor solutions that improve the global performance index. Once a stochastic method has been created, it is usually versatile: it does not have references to the physical problem from which it was born and can therefore be applied in many fields. Stochastic methods offer robustness to optimization process. They are best suited for discrete problems, but they can be tailored to continuous problems as well. They are able to find the optimal solution provided that their variable population is smaller than that used in the previous methods, because their computational cost is extremely high. Stochastic search techniques are grouped into two major subclasses: Evolutionary Algorithms (Evolutionary Strategies - ES - and Genetic Algorithms - GA), and Simulated Annealing (SA). ES and GA emulate nature's evolutionary behavior by use of natural selection principles and biologically inspired operations, while SA mimics the thermodynamic evolution process of annealing a material.

The choice of which optimization technique to use will depend upon many factors, among which we have:

- Whether the objective function is one or many-dimensional.
- Whether the objective function can be differentiated efficiently.
- How noisy the objective function is.
- Which is the number of design variables.
- Which is the type of design variables (real/integer, continuous/discrete).
- Whether the objective function is linear or nonlinear.
- Whether there are equality or inequality constraints.
- Whether there are discontinuous feasible spaces or not.
- Whether the initial solution is feasible or infeasible.
- How onerous the simulation code runtime is.

Therefore, before applying any of the aforementioned techniques, it is important to understand:

- Whether and why it is appropriate to use a particular method.
- The basics of how the method works.
- Why the method might fail.

while after applying the selected algorithm, it is necessary to understand what the obtained results mean (numerically and physically).

A multiobjective problem has conflicting objectives, and involves several disciplines, and therefore it has more than one optimal solution. All the considerations seen so far are generic mathematical tools. The main difference between a single objective optimization problem and a multiobjective optimization problem is that in the first the goal is to find a single optimal solution, while in the second the goal is to find a set of optimal solutions. There are two fundamental approaches to multiobjective optimization: scalarization approaches, in which the designer's preferences are included upfront in the optimization problem, and Pareto approaches, in which the designer's preferences are included a posteriori in the selection of one optimal solution.

Possible methods for solving multiobjective problems are weighted-sum approach, where each objective is multiplied by a strictly positive weight λ_i chosen such that the sum of all weights equals unity, and then the weighted sum problem is solved by linear programming, and Pareto-filter approach, where an approximation of the Pareto front is obtained by evaluating dominatedness or non-dominatedness of solutions found, for example, with an exploration of the design space¹³

There are also methods for explicit calculation of the Pareto front. These methods must have the ability to capture all Pareto points, managing scaling mismatch between objectives, and achieving an even distribution of solutions on the front with an even distribution of the input parameters. Some of them are Normal Boundary Intersection (NBI) and Adaptive Weighted Sum Method (AWSM).

Of course, there is also the possibility of using heuristic methods for multiobjective problems. One example are multiobjective genetic algorithms, where the ranking of population is made with a Pareto criterion, without assigning a priori preference to certain objectives.

Over the last two decades, multidisciplinary design methodologies have acquired greater maturity, and are now capable of facing new technological problems. The features offered by software optimization tools allow a larger number of configurations and design alternatives to be compared in less time than in traditional approaches. The automatic management of design variables can therefore also be exploited to analyze the behavior of highly non-linear systems, while ensuring the simulation of a large number of configurations. Multidisciplinary and multi-objective optimization is now a promising method that can be used to manage increasing system complexity, to support product development, considering program objectives and constraints, and to allow the identification of optimal solutions among various alternatives.

The application of multidisciplinary analysis and optimization techniques to the research conducted in Thales Alenia Space was carried out in three steps. The first step was a case study referred to the ABB prototype, and taking into account thermal and structural characteristics, the second step was a case study regarding the STEPS prototype, and still taking into account thermal and structural

¹³Let \mathbf{J}^1 and \mathbf{J}^2 be two objective vectors. Then \mathbf{J}^1 (weakly) dominates \mathbf{J}^2 if and only if $\mathbf{J}^1 \geq \mathbf{J}^2$ and $\mathbf{J}^1 \neq \mathbf{J}^2$ ($J_i^1 > J_i^2$ for at least one i). And \mathbf{J}^1 strongly dominates \mathbf{J}^2 if and only if $\mathbf{J}^1 > \mathbf{J}^2$ ($J_i^1 > J_i^2$ for each i). All points on a Pareto front are non-dominated.

performances, while the third case was a re-engineering study of the ABB prototype taking into account thermal, energy budget and control aspects. Referring back to the general description of optimization techniques, these are all studies conducted in a distributed analysis approach, i.e. different modules carry out the analysis at subsystem/discipline level, while an integrated environment takes care of the overall optimization.

3.1.3 ABB loop

The ABB case study dealt with a demonstrative optimization loop including mechanical and thermal analysis. The discipline level modeling and analysis was performed by means of Altair HyperMesh (generation of Finite Element Models FEMs from CAD geometries, pre-processing and handling of meshes) and Nastran for the structural aspects, and NX TMG Thermal Analysis (Transient and steady state solution of conductive, convective, radiative, and fluidic heat transfer phenomena) for the thermal aspects. The overall optimization was conducted thanks to iSight (automatic handling of design process, integrating external analysis tools, DoE and optimization algorithms, and post-processing analysis tools).

The complete ABB model was simplified in different ways, according to the aspects that were more relevant to the different disciplines.

Under a structural point of view, only the bulky elements were considered, i.e. the support panel, the DC/DC converter and its clamping system. The support panel has been modeled as a sandwich configuration with shell skin elements (elastic modulus = 38 GPa), and an orthotropic equivalent honeycomb based on hexa elements with compression modulus = 28.15 MPa and shear modulus = 247 MPa). The structural loads taken into consideration were the accelerations induced by launch with a Soyuz vehicle: in particular, a value of $5g$ has been fixed along a direction normal to the ABB panel (through-the-thickness direction). The panel itself has been considered constrained with rigid joints on all edge nodes of the lower skin. The loads have been computed and applied before the setup of the optimization problem and, given the small variations foreseen, left fixed throughout the loop.

To enable the variation of geometries during the optimization loop, in particular to change the thickness of the core layer, a morphing technique has been used: morphing perturbation vectors obtained through Altair HyperMorph are combined with a morphing perturbation value (changed during optimization) to compute the modified mesh nodal positions, which are subsequently included in the bulk data file before launching the structural analysis.

Under a thermal point of view, on the other hand, the relevant components of the ABB model are the motherboards ($k=0.3$ W/m/K), with DC/DC converters and heaters, and the C/C panel. In particular the panel is composed by two skins connected to each other by a thermal coupling in the transverse direction ($k_z=1.5$ W/m/K). Load heaters and top skin are connected with a heat transfer coefficient of 2000 W/m²/K. The Araldite adhesive bondings between heaters and motherboards, motherboards and top skin, and DC/DC converters and top skin have also been taken into consideration ($k=0.35$ W/m/K). The effective emissivity of the MLI (radiative exchange with the thermal vacuum chamber) has been set to

Table 3.2: Dependence of MLI conductivity on temperature.

T (°C)	λ (W/m/K)
-180	0.0205
-140	0.0287
-100	0.0398
-50	0.0592
0	0.0865
50	0.1248
80	0.1545
100	0.1777
140	0.2336
180	0.3049

Table 3.3: Characteristics of polyimide motherboards.

Density	5508.85 Kg/m ³
Young modulus	4826 MPa
Poisson ratio	0.1
CTE	25 ppm/°C
Conductivity	0.3 W/m/K

0.05, while those of the top, bottom skin and panel's sides has been fixed to 0.43 (radiative exchange between top skin and MLI, and between bottom skin or panel's sides and vacuum chamber). The effective emissivity of polyimide motherboards has been set to 0.77 (radiative exchange between MBs and MLI). The temperature boundary conditions chosen for the vacuum chamber are those measured during the test campaign, roughly ranging from -158 to -170 °C.

The thermal properties of multilayer insulations are described in Table 3.2, while the characteristics of the polyimide motherboards are listed in Table 3.3.

The loop consisted of a structural section, which takes care of mechanical analysis and morphing, a thermal section, which tailors the thermal properties taking into account possible variations of geometry due to the structural section, and then runs the thermal analysis, and a post-processing session, which computes performance indexes like average temperatures. Structural parameters were the skin thickness *thick* and the morphing modulus *magn*, while performance was evaluated on the maximum x,y, and z displacements extracted from the bulk file. Thermal parameters were the honeycomb conductivity (which was updated during every cycle taking into account the structural morphing), and the power dissipated by the heaters *Main_H*, while performance was evaluated on temperatures detected in strategic points (DC/DC converter, motherboards, and several points on the skin), and on ΔT across the panel's thickness.

Table 3.4: Design of experiment for the ABB demonstration MDO loop.

Parameter	Min	Baseline	Max
<i>magn</i>	-25%	1	+20%
<i>Main_H</i>	-25%	10 W	+25%
<i>thick</i>	-25%	0.85 mm	+25%

At first, a design of experiments was conducted, to highlight the effects and possible couplings of the different parameters on the overall system performance. It was an Optimal Latin Hypercube study with three parameters, as shown in Table 3.4.

As shown by the results depicted in Figure 3.6, this DoE highlighted that the power dissipated by the heaters and the thickness of the skins, strongly influences the temperature difference between panel's faces. The variations in core thickness has little influence on the ΔT across the thickness. The thickness of the skins influences both thermal conductivity and panel deflection. Increasing *magn*, i.e. increasing the distance of the elements able to bear flexure from the neutral axis, the structure becomes stiffer, and the same happens increasing the thickness of the skins *thick*. Increasing the power dissipated by the heaters obviously increases both the temperatures in the various test points and the ΔT between faces. The core thickness heavily affects the thermal conductivity in the transverse direction.

	ADADDX	ADADSX	MBDX	MBSX	PAN_delta
Main_H	91.88	91.88	91.89	91.88	88.55
Main_H-thick	0.092	0.09	0.09	0.088	1.383
magn	0.182	0.181	0.173	0.172	3.42
magn-Main_H	0.115	0.118	0.113	0.117	0.814
magn-thick	0.052	0.052	0.053	0.052	-0.014
thick	0.378	0.378	0.363	0.362	5.81
	lambda	thick_tot	x_max	y_max	z_max
Main_H	-0.003	0.006	1.81	1.38	0.5
Main_H-thick	0.013	-0.017	-1.92	-2.55	-0.798
magn	-35.91	36.97	0.034	-0.3	-0.284
magn-Main_H	0.05	-0.05	-1.427	-0.375	-0.284
magn-thick	1.385	-0.03	0.348	0.09	1.682
thick	-61.05	62.86	-61.79	-66.08	-73.04

Figure 3.6: Percent correlation between study parameters and system responses according to the DoE.

After the design of experiments, two optimization loops were conducted. The first one was a thermal optimization loop, aimed at finding the best compromise between dissipated power and panel thicknesses. The second one was a thermo-structural loop. Different optimization techniques were used in order to evaluate their behavior and performances.

The thermal loop was set up to find the optimal power dissipation and panel

Table 3.5: Thermal optimization for the ABB case study: ranges for the design variables.

Parameter	Min	Baseline	Max
<i>Main_H</i>	10.0 W	10.0 W	50.0 W
<i>thick_try</i>	15.0 mm	19.5 mm	30.0 mm

Table 3.6: Thermo-mechanical optimization for the ABB case study: ranges for the design variables.

Parameter	Min	Baseline	Max
<i>Main_H</i>	9 W	10 W	30 W
<i>magn</i>	0.5	1.0	2.0
<i>thick</i>	0.5 mm	0.85 mm	1.5 mm

thickness, according to the constraint of keeping the temperatures on the motherboards and on the skins, near the DC/DC converters around 20°C and pursuing the goal of minimizing temperature difference between the skins. The parameters were bounded as per Table 3.5.

Three different algorithms have been used: Multifunctional Optimization System Tool (MOST, gradient based), Non-dominated Sorting Genetic Algorithm - II (NSGA-II, evolutionary stochastic), and Adaptive Simulated Annealing (ASA, stochastic physically inspired).

According to the MOST algorithm, the optimal solution is to keep the power at 11.96 W and minimize the thickness (it saturates against the lower boundary, which is a sign that a second goal or constraint, maybe from another discipline, is needed to steer the selection of this parameter). Apart from the initial iterations, the solutions converge quite smoothly to the final result.

According to NSGA-II, the optimal dissipated power is 11.78 W and the optimal thickness 16.34 mm. The evolution of the solution between subsequent runs is much more noisy with respect to the MOST algorithm, and this is a typical characteristic of heuristic methods.

According to the ASA algorithm, two optimal points were identified: one with *Main_H* = 12.29 W and *thick_try* = 18.93 mm, and the other one with *Main_H* = 11.97 W and *thick_try* = 28.31 mm. The first solution allows slightly improved radiative performance and weight, while the second one has slightly better temperature distribution with respect to the constraints.

After the first thermal optimization, a combined thermo-mechanical loop was implemented. The design variables, constraints and objectives are described in Tables 3.6, 3.7, and 3.8.

The optimization has been conducted with two different algorithms: MOST and NSGA-II. The respective optimal results are compared in Table 3.9.

Looking at the results of both optimization processes it is immediately evident

Table 3.7: Thermo-mechanical optimization for the ABB case study: constraints for the computed temperatures.

Parameter	Min	Max
ADADXSX	-55 °C	100 °C
ADADDX	-55 °C	100 °C
MBSX	0 °C	70 °C
MBDX	0 °C	70 °C

Table 3.8: Thermo-mechanical optimization for the ABB case study: objectives.

Parameter	Criterion	Weight
PAN_delta	minimize	1
x_max	minimize	1
y_max	minimize	1
z_max	minimize	1

Table 3.9: Thermo-mechanical optimization for the ABB case study: objectives.

	MOST		NSGA-II	
Iteration n.	46		141	
Objective	2.1482		2.1469	
Penalty	0.0		0.0	
Objective&Penalty	2.1482		2.1469	
Main_H	9.1322	W	9.1454	W
thick	1.0068	mm	0.9628	mm
magn	0.5		0.527	
ADADDX_mean	1.052	°C	1.142	°C
ADADXSX_mean	1.052	°C	1.142	°C
MBDX_mean	0.002	°C	0.092	°C
MBSX_mean	0.002	°C	0.092	°C
PAN_delta_mean	2.129	°C	2.127	°C
x_max	9.077 e-4	mm	9.495 e-4	mm
y_max	0.001488	mm	0.001556	mm
z_max	0.016822	mm	0.01748	mm

that the solutions are almost identical, but the NSGA-II algorithm needed ~ 7 hours of runtime, while the MOST algorithm concluded the search in 2.5 hours.

With respect to the monodisciplinary thermal optimization, the thermo-mechanical study pushed the optimal solution towards less extreme values of design variables and system response: this is because, in the combined domain, there are two contrasting requirements, i.e. to reduce the temperature difference between the two skins, which calls for a thickness reduction in order to minimize the thermal resistance, and to reduce structural deformation, which requires increased thickness (mainly, increased skin thickness). Therefore, since objective weights were identical, a compromise solution is the obvious outcome. In fact, deformations are decreased by an increase in the *thick* parameter, which goes from 0.85 mm to 0.97 mm. It is very interesting to notice that, even if the morphing variable *magn* has an effect on the total thickness, its value is not increased: it has been reduced from 1 to 0.527, and this is because *magn* mainly influences the core thickness, which has a much smaller influence on mechanical response, and can therefore be diminished in favor of better thermal performances. However, the overall thickness is greater than the one foreseen by the purely thermal optimization (~ 22.22 mm versus 16.5 mm). An increased thickness means a lower rejection capability, because of an higher temperature difference between the faces, but this problem has been overcome by lowering the dissipated power. Indeed, the final ΔT obtained with this approach is even lower than the one foreseen in the thermal optimization.

This activity gave a good overview of issues and good practices related to modeling and simulation for the purpose of performing a multidisciplinary optimization. It was also useful in testing different techniques for the design of experiment and optimization phases. Moreover it showed clearly the balancing behavior introduced by an optimization process in presence of several conflicting drivers. Incidentally, the work was also a good demonstrator of the necessity of human supervision to make educated choices during the process. Additional details on this activity can be found in [50].

3.1.4 STEPS loop

Similarly to the previous exemplification, the demonstration loop based on the STEPS case study took into consideration thermal and structural aspects.

The software tools used to perform modeling and analysis are NX 6 CAD and Nastran for the structural domain, and NX Thermal Analysis for the thermal domain. The overall optimization process integration has been carried out thanks to iSight.

The model is focused on a quarter of the complete STEPS SDA demonstrator: a 220 mm x 310 mm x 21.5 mm area of the support panel equipped with one 300 mm x 210 mm smart skin module.

The thermal model takes into account the aluminium sandwich support panel, the motherboard with its embedded heaters and the external load heaters placed on the MB itself. In analogy to what has been done with the ABB study, the sandwich core is not modeled through actual element, but instead replaced by a conductive coupling ($k=2$ W/m/K) between the top and bottom skins. The skins have been modeled as AA 2024 - T81 Clad sheets with an initial thickness of 0.35 mm and

Table 3.10: Mechanical properties of the panel's skins.

Material designation	Al 2024-T81 clad
Young modulus (tension) E	72390 MPa
Young modulus (compression) E_c	73770 MPa
Poisson ratio ν	0.33
Density ρ	2768 Kg/m ³
Yield stress (tension) σ_y	372 MPa
Ultimate stress (tension) σ_u	427 MPa
Yield stress (compression) σ_{cy}	379 MPa
Ultimate stress (shear) σ_{cu}	262 MPa

an emissivity of 0.77, while the core is made of 3/16 - 5056 - 0.0007 aluminium honeycomb.

Motherboards' material is DuPont Pyralux AP9121, and they have been modeled as 0.15 mm thick layers with an emissivity equal to 0.77. Its characteristics are listed in Table 3.3.

The MLI has been modeled as a 5 mm thick shell with emissivity $\varepsilon = 0.05$ and conductivity dependent on temperature as per Table 3.2.

The coupling between motherboards and heaters and between motherboards and top skin have been described with a conductivity $k = 0.35$ W/m/K. The effective emissivity for the radiative exchange between motherboards and MLI has been fixed to 0.0493, while the effective emissivity between heaters and MLI has been assumed equal to 0.0256.

Regarding the boundary conditions, the small panel has been considered surrounded by a sink emispase at -180 °C, added to set up a radiative coupling with the lower skin (emissivity equal to 0.8), and by a MLI surface, included to impose a radiative exchange with the upper skin.

The load conditions for the additional heaters representing external loads have been assumed in a mildly asymmetric configuration: 10 W on the left side and 6 W on the right side.

The structural model takes into account the aluminium supporting panel, with the skins modeled through quad shell elements, and the core described with hexa solid elements. The reference materials and their properties are summarized in Tables 3.10 and 3.11.

Like in the ABB case, the load and boundary conditions have been created under the hypothesis of accommodating the structure within a Soyuz launcher. A 5g acceleration applied along the through-the-thickness direction determines the structural loads. The structure has been considered constrained by eliminating all DOFs for the edge nodes of the bottom skin.

After definition and validation of the independent disciplinary models, the activity shifted toward the exploration of the design space and the optimization of the configuration.

The first attempts were made on the thermal aspects, including the physical

Table 3.11: Mechanical properties of the panel's core.

Material designation	3/16 - 5056 - 0.007
Density ρ	32.1 Kg/m ³
Compressive modulus E_c	310.1 MPa
Shear modulus ribbon (L) direction G_L	186.0 MPa
Shear modulus width (W) direction G_W	89.6 MPa
Compressive strength σ_c	0.827 MPa
Allowable shear stress (L direction) σ_{sL}	0.723 MPa
Allowable shear stress (W direction) σ_{sW}	0.345 MPa
Cell size S	4.763 mm

Table 3.12: Ranges for the thermal design variables during DoE.

	Min	Baseline	Max
Heater_Up_Dx	0 W	0 W	7 W
Heater_Up_Sx	0 W	0 W	7 W
Heater_Down_Dx	0 W	0 W	7 W
Heater_Down_Sx	0 W	0 W	7 W
Thick_Skin_Up	0.2 mm	0.35 mm	0.5 mm
Thick_Skin_Down	0.2 mm	0.35 mm	0.5 mm

simulation with TMG and the extraction of significant parameters through a post processing code for the computation of minimum and maximum temperatures on the motherboard and the associated ΔT . A design of experiments was run, with the settings described in Table 3.12 to determine the relative influence of different parameters on temperature uniformity.

Given the asymmetric load case, with a higher dissipation on the left side, the heat load exerted by the control heaters on the right side has the main effect on ΔT reduction. The influence of control heaters on the left side is slightly lower, and the effect of skin thickness variation, although sensible, is the least relevant. Subsequently, an optimization study aimed at minimizing the temperature disuniformity was carried out by means of the Downhill Simplex algorithm. This analysis highlighted the fact that the minimum achievable ΔT is around 10 °C, as shown in the optimum configuration summarized in Table 3.13.

Then, a full thermo-mechanical optimization loop was conducted, with the goals of minimizing the mass and the ΔT on the motherboard. The single iteration flux includes a preliminary purging of data files, followed by the adjustment of input values inside a data file that can be included in the Nastran input file. At this point the mechanical analysis can be performed, and the results can be examined to extract the information regarding deflection and mass. Subsequently, the thermal analysis is launched and data regarding min-max temperatures and

Table 3.13: Optimal dissipations according to the Downhill Simplex algorithm.

Parameter	Value
Heater_Up_Dx	4.23 W
Heater_Up_Sx	0.36 W
Heater_Down_Dx	2.96 W
Heater_Down_Sx	0.66 W
Thick_Skin_Up	0.45 mm
Thick_Skin_Down	0.42 mm
Delta_T	10.10 °C

Table 3.14: Constraints for the thermo-structural optimization.

Quantity	Lower value	Upper value
Temp_Max	10 °C	50 °C
Temp_Min	-5 °C	30 °C
Displacement_Z	-	$10 \cdot 10^{-4}$

ΔT are obtained.

The constraints fixed for the optimization were not very stringent and are summarized in Table 3.14.

The baseline and boundary values for the design variables have been chosen on the basis of the aforementioned thermal optimization, and are described in Table 3.15.

The optimization, conducted by means of the Downhill Simplex method, gave the optimal solution summarized in Table 3.16. Given the fact that the starting baseline values were skewed toward higher temperatures on the upper side of the test article, the final solution inherits the same asymmetry. Considering the similarity of this result with the one identified taking into account just the thermal domain,

Table 3.15: Design variables: baseline and boundary values.

	Lower	Baseline	Upper
Heater_Up_Dx	0 W	4 W	7 W
Heater_Up_Sx	0 W	0.5 W	7 W
Heater_Down_Dx	0 W	3 W	7 W
Heater_Down_Sx	0 W	0.3 W	7 W
Thick_Skin_Up	0.2 mm	0.45 mm	0.5 mm
Thick_Skin_Down	0.2 mm	0.4 mm	0.5 mm

Table 3.16: Optimal solution according to the Downhill Simplex algorithm.

Parameter	Value
Heater_Up_Dx	4.39 W
Heater_Up_Sx	0.44 W
Heater_Down_Dx	2.76 W
Heater_Down_Sx	0.21 W
Thick_Skin_Up	0.49 mm
Thick_Skin_Down	0.40 mm
Delta_T	10.18 °C
Mass	0.21 Kg
Temp_Max	32.60 °C
Temp_Min	22.42 °C
Displacement_Z	$4.80 \cdot 10^{-4}$

it is positive to see that the algorithm is able to steer the design in the correct direction even in presence of contrasting drivers. In particular, this case study was also interesting in exploring the role of user preferences when introduced a priori in the analysis. For a thorough description of this activity refer to [51].

3.1.5 ABB re-engineering loop

After exploring the potentialities and characteristics of multidisciplinary analysis and multiobjective optimization with the ABB and STEPS demonstration loops, a more articulated study has been performed with the aim of re-engineering the ABB concept to pave the way for the ROV-E smart skin architecture. For this reason, the work dropped the structural aspects and focused on thermal, energy budget, and control aspects.

The first part of the work concerned the definition of the thermal model. The working environment selected for the definition of the model was Matlab-Simulink: this choice was motivated by the powerful modeling and computing environment offered by the tool workspace and, in particular, by the possibility of an easy integration of different thermal control strategies. In fact, the previous experiences had strongly highlighted the fact that an appropriate level of modelization has a great influence on the ease and significance of the optimization studies. Matlab-Simulink allowed a sufficient level of detail in the model without excessive computational burden, but at the same time added a precious flexibility in handling the addition of control logics to the thermal model.

When setting up the thermal model, attention was primarily paid to the determination of the energy consumption of the thermal control heaters, and of the thermal gradient on the panel. As far as the energy requirements for a particular heater arrangement and the operational requirements of the electronics are concerned, it is helpful to understand which configuration allows the associated consumption to be reduced as much as possible. In fact, the resources available

on board the space system are limited, and an appropriate thermal control design strategy should consider their savings. An analysis of the temperature distribution along the panel is an important aspect concerning the thermal stresses to which the structure is continuously subjected. The possibility of reducing the thermal gradients would improve the integrity of the various elements of the system under consideration.

A simple but flexible lumped parameter model has been preferred, and the laws associated with the electrical analogy of thermal systems have therefore been considered. The three-dimensional model was essentially divided into 2 layers representing the upper and the lower skins of the multifunctional panel, respectively. The model is made of 160 thermal nodes for the upper skin and an equal number for the lower skin. This value represents a proper balance between the representativeness of the thermal description for the real ABB panel and easiness of modeling. In addition, other nodes necessary for the definition of the boundary conditions have been included. These were needed to model the radiative heat transfer between the nodes of the lower skin and the surrounding environment (namely, the cold shrouds of a thermal vacuum chamber). The radiative heat transfer phenomenon with other elements has not been considered for the upper surface, as the presence of an MLI blanket and, therefore, of a negligible heat outflow affecting these regions, has been assumed.

Each node identifies a specific portion of the material. In the discretization scheme that has been adopted, part of the core material is assigned to the nodes of the upper skin and the other portion is instead assigned to the lower skin. The thermal resistance values between the individual nodes are calculated from the real material properties, with reference to the values obtained experimentally during laboratory tests. The same procedure was adopted for the evaluation of the nodal thermal capacity.

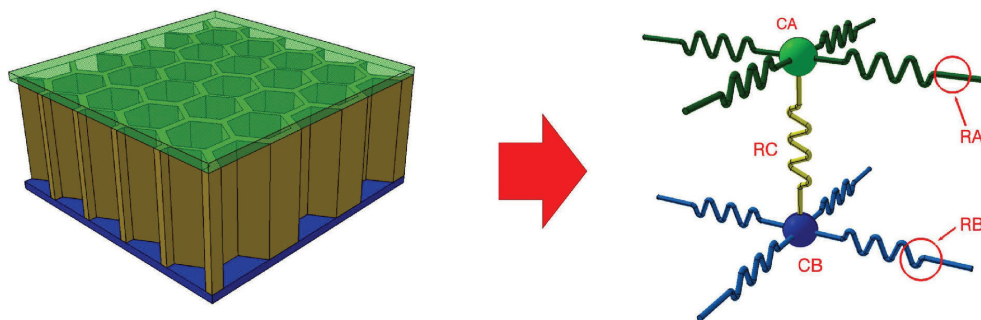


Figure 3.7: Discretization of a single element starting from the geometric representation.

The nodes of the upper skin are indicated in green in Figure 3.7, while the ones belonging to the lower region are represented in blue. Consistently with the above mentioned description of the model, the quantities involved in the modeling activity have also been identified in the representation. Figures 3.8 and 3.9 show 3D-CAD representations of the lumped model, and highlight the implementation of the previous discretization on the whole system.

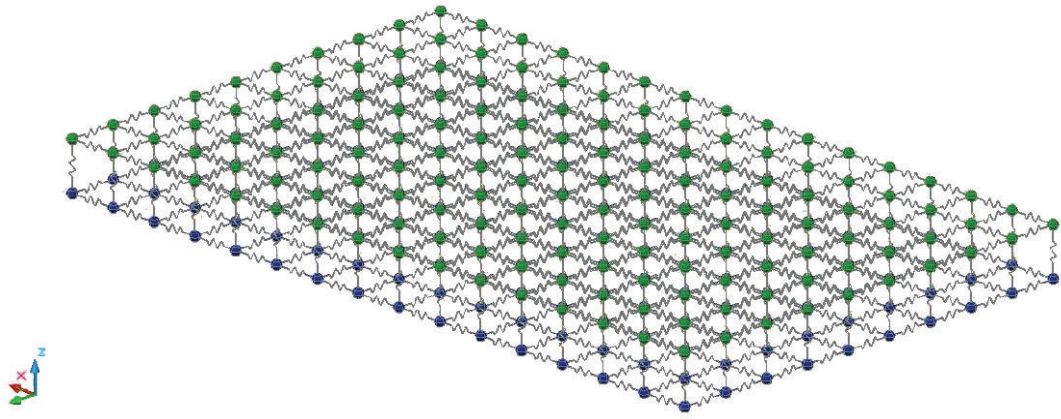


Figure 3.8: 3D thermal model layout: isometric view.

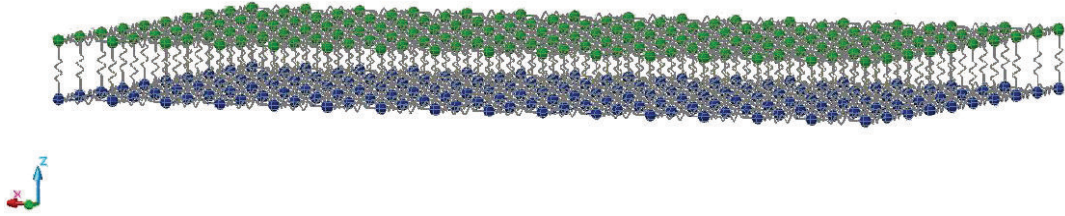


Figure 3.9: 3D thermal model layout: side view.

Each of the aforementioned thermal resistances was computed with the expression:

$$R_T = \frac{L}{\lambda \cdot S}$$

where L is the geometrical distance between the nodes characterized by heat exchange, λ is the thermal conductivity (under the assumption of material homogeneity along the conductive path), and S defines the transverse surface area for the heat flux. The calculation of the involved quantities refers to the three-dimensional representation shown in Figure 3.10, which is a schematic visualization of one of the basic elements into which the structure is discretized. The picture, in particular, schematically represents a portion of the real geometry of the structure, and, for convenience, the upper skin has not been visualized. In order to have an idea of the resolution of the model, the base of the depicted element represents a square surface portion of the panel with a 5 cm side. In the same figure, the cross-section, which is responsible for the passage of the heat flux in the transverse direction, is indicated with a black solid line: this section is associated with the definition of the thermal resistance along the core.

The knowledge of the thermal resistance value between two nodes was used to define the power flow between these nodes, once the temperature difference is known.

$$q_T = \frac{1}{R_T} \cdot (T_i - T_j)$$

All the heat flows in the system have been calculated by implementing the same relationship in the Simulink environment (by building a block diagram). After the definition of the heat fluxes, a set of energy balance equations has been implemented:

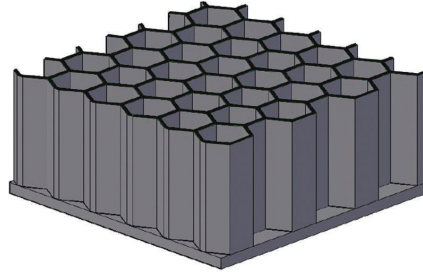


Figure 3.10: Transverse surface area for the heat flux through the core.

starting from the known heat fluxes, it has been possible to define the time rate of the change in the nodal temperatures, which also depends on the associated heat capacity. As far as the heat capacity is concerned, the model used the following expression:

$$C_T = m \cdot c_{SP}$$

where m identifies the mass of the sub-volume, while c_{SP} identifies the specific heat that characterizes the material. Then, according to the heat capacity values associated with each node, it has been possible to write the relation referring to the balance equations, which is given in the following expression:

$$\sum q = C_T \cdot \frac{\partial T}{\partial t}$$

The flux is positive when it flows into the node. Again, in a similar way to what has been done for the definition of the flows, it was necessary to introduce a balance equation for each node in the system. In addition to the previous relations, the equations associated with the heat exchange through radiation, which were necessary for the characterization of the boundary flows, have also be considered:

$$q_{boundary} = \frac{1}{R_D} \cdot (T_i^4 - T_j^4)$$

where R_D identifies the thermal resistance for heat transfer through radiation, that is:

$$R_D = \frac{1}{A \cdot \sigma \cdot \varepsilon}$$

where A refers to the surface involved in the radiation exchange, σ identifies the Stefan-Boltzmann constant and ε is the emissivity. The radiative view factors were considered equal to unity throughout the model. As for the applied model, the thermal resistances associated with the structure of the panel were all of the same magnitude, although they could in principle be differentiated from each other when considering other structural configurations. For the same reason, the diffusive heat capacities associated with the nodes of the panel were all taken equal to each other, but they could be distinguished on the basis of the design requirements. The definition of the numerical characteristics of the in-plane resistances refers to the

following values:

$$R_A = R_B = \frac{L}{k_x \cdot S} = \frac{L}{k_y \cdot S}$$

$$R_A = R_B = \frac{0.05}{40 \cdot 0.00025} = 5K/W$$

where the values for conduction path length L and flow section S perpendicular to the heat flux were estimated on the basis of the discretization adopted in the model. The following value has been estimated for the heat transfer along the core:

$$R_C = \frac{L}{k_z \cdot S} = \frac{0.02}{1.5 \cdot 0.00072} = 18.5K/W$$

In this case, the flow section and the length of the conductive path were also evaluated from the physical discretization. The values of the thermal capacities of the diffusive nodes have been computed according to the following relations:

$$C_A = C_B = V_A \cdot \rho_A \cdot c_{SP}$$

$$C_A = C_B = 0.05 \cdot 0.05 \cdot 0.005 \cdot 1700 \cdot 710 = 15.1J/K$$

The estimated values could undergo changes, if the experimental results on a real panel were taken into account in order to offer greater validity to the model and simulations.

After the definition of the geometry and material characteristics of the model, the following phase was represented by constructing the model in the Simulink environment. The representation of the whole scheme of the model is shown in Figure 3.11 for clarity reasons.

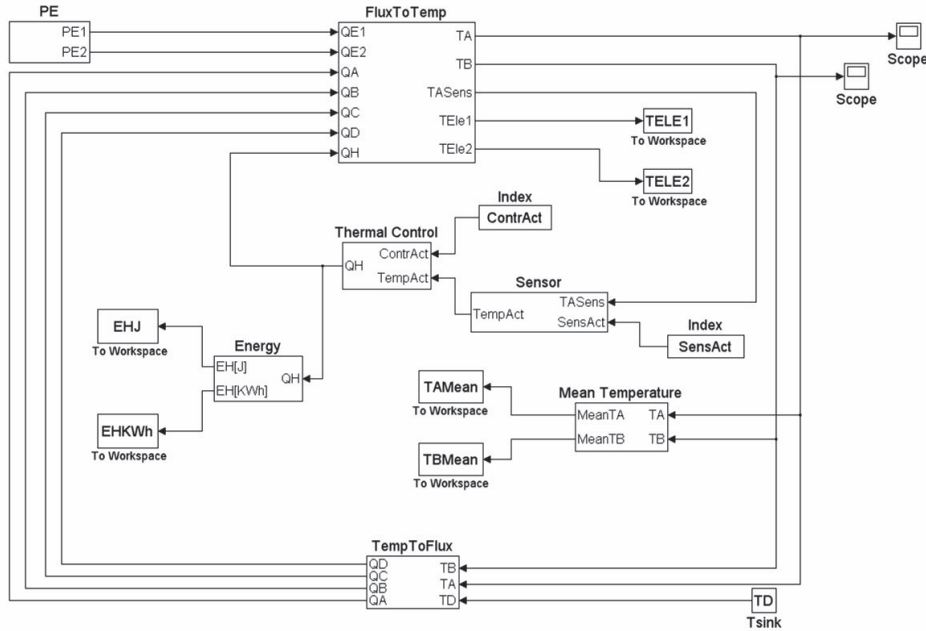


Figure 3.11: Complete Simulink model.

The first step was to implement the equations required to develop the thermal flows among the different nodes. For greater readability, the equations have been

divided into a) flows related to the upper skin, b) flows related to the lower skin, c) flows related to the core, and d) those related to the boundary conditions. Thus, the obtained heat fluxes can be used for the construction of the balance equations. For the sake of clarity, the time evolution of the temperature, obtained from the definition of the flows, has been divided into the upper skin (identified with the term TA) and the lower skin (identified with the term TB). Apart from the flows associated with the heat exchange between the nodes, and those associated with radiative boundary diffusion, it was also possible to add the flows related to thermal control. Each node in the upper skin could potentially be subject to an incoming heat flux from the control subsystem (in practice, these heat loads are realized by incorporating heating elements such as Kapton heaters). When there was no thermal control on a specific node, the thermal control flux value was set equal to zero. In this case, it was necessary to consider the fluxes related to the boundary radiative exchange with the thermal sink for the nodes discretizing the lower surface, but there was no power input due to the heat dissipative elements. The presence of the two motherboards, characterized by a certain heat generation, was also considered on the portions of the upper skin relative to the electronics housing. There were two different heat load profiles and the two electronic boards could be activated independently in order to simulate different operation configurations. This is the reason why it is possible to detect the presence of the external heat load PE, among the signals introduced into the system: this PE load identifies the value of the heat flux associated with operation of the motherboards. In this case, the input signals were modeled with piecewise linear functions in order to describe the time evolution of the thermal power generated by the two electronic units, respectively. Their patterns represented the boundary conditions of the optimization problem.

The two blocks relevant to the implementation of the balance equations, in Figure 3.11, are *FluxToTemp* and *TempToFlux*. The first subsystem, *FluxToTemp*, is essentially the block that allows the heat fluxes to pass to the temperatures of the different nodes, depending on the balance equations. QA, QB and QC represent the flows that affect the top skin, bottom skin, and core, respectively. Finally, QD is the flow of radiative heat exchange with the outside, while QH is the flow associated with the thermal control logic (heat flows related to the heaters only operate on the A skin, consistently with the fact that the thermal control elements are located on the top surface). The output associated with the term *TASens* represents the vector of temperatures that can potentially be adopted to manage the control law. In the present case, this term is assumed to be equal to the TA vector of temperatures. The second subsystem, *TempToFlux*, computes the heat fluxes generated by a given temperature distribution. It receives the temperatures of the two skins and the boundary condition (which is only interfaced with the lower radiative surface) as input. The outgoing signals are represented by the diffusive terms present in the upper (A) and lower (B) skins, and through the core (C), as well as by the radiative terms related to the boundary conditions (D). It is possible to easily manage the boundary temperature of the system in the model by operating on the TD block. This also makes it possible to assess the correctness of the model through a comparison of the obtained results with experimental data from the thermal-vacuum tests. In the present case, the simulations were conducted assuming a constant temperature of the surrounding space.

The modeling of the temperature control subsystem was implemented individually with a particular subsystem, named *Thermal Control*. In this case, the temperatures of the upper surface are considered as input quantities, as they can potentially be taken as a reference to manage the control law. The output wattage is controlled on the basis of the thermal control strategy that is used, monitoring the nodal conditions. It is generally possible to develop controls in which the temperature used for thermal management is detected in other nodes, and not necessarily in the same ones into which the thermal flux is introduced by the heaters. It is possible to introduce different control strategies in the thermal control block, in order to identify those parameters or, more generally, that law which provides the greatest benefits. After defining which nodes are checked in the model (i.e. “activated” or, in other words, selected for thermal control behavior), and which control law should be adopted (this information is contained in the *ContrAct* variable), these data are passed to the control subsystem. The vector signal of load heat fluxes QH is generated, in this subsystem, according to the activation requirements of the TCS, and depending on the temperature values of the “activated” nodes. As previously mentioned, it is usually possible to define several thermal control laws inside this thermal control block, the activation of which can be managed through the evaluation of other parameters. In this way, it is possible to explore different TCS solutions, and to evaluate the most convenient one.

The objective function used in the optimization process was based on an estimate of the energy absorbed by the thermal control law, coupled with a certain heater configuration (and in compliance with the operative temperature requirements of the electronics). A subsystem that receives the vector of all the possible thermal control flows (of which only active ones will have non-zero value) as input was introduced to estimate the energy consumption associated with that particular control solution. The integration of the power values made it possible to compute the energy consumed to perform this control.

Besides the computation of the total energy consumption, another result of interest is that of the acquisition of the temperature distribution on the upper surface of the model. In the simple example shown in Figure 3.12, the represented temperature distribution (referring to the upper surface of the panel) is obtained under the assumption of considering the first 10 nodes of the long side of the upper skin operative. The nodes corresponding to the electronics are considered disabled. In this way, the thermal load inputs are only those related to the heaters. The amount of heat flow is such that the panel temperature is increased until the inflow equals the heat lost through radiation, in the steady regime.

A most relevant step of this analysis was the validation of the model, which was needed to confirm that model responses were consistent with the available experimental data. The management of all the parameters included in the model was able to cope with the uncertainty of many data, e.g. of the values of the thermal conductivity. The latter, in fact, is not necessarily known accurately and its evaluation depends to a great extent on manufacturing conditions, and on its integration with other materials. From this point of view, model validation is also useful to confirm the reliability of the data regarding the material properties. A fundamental verification of the basic physical laws was obviously necessary: the relevant simulations were characterized by the absence of thermal loads of any kind,

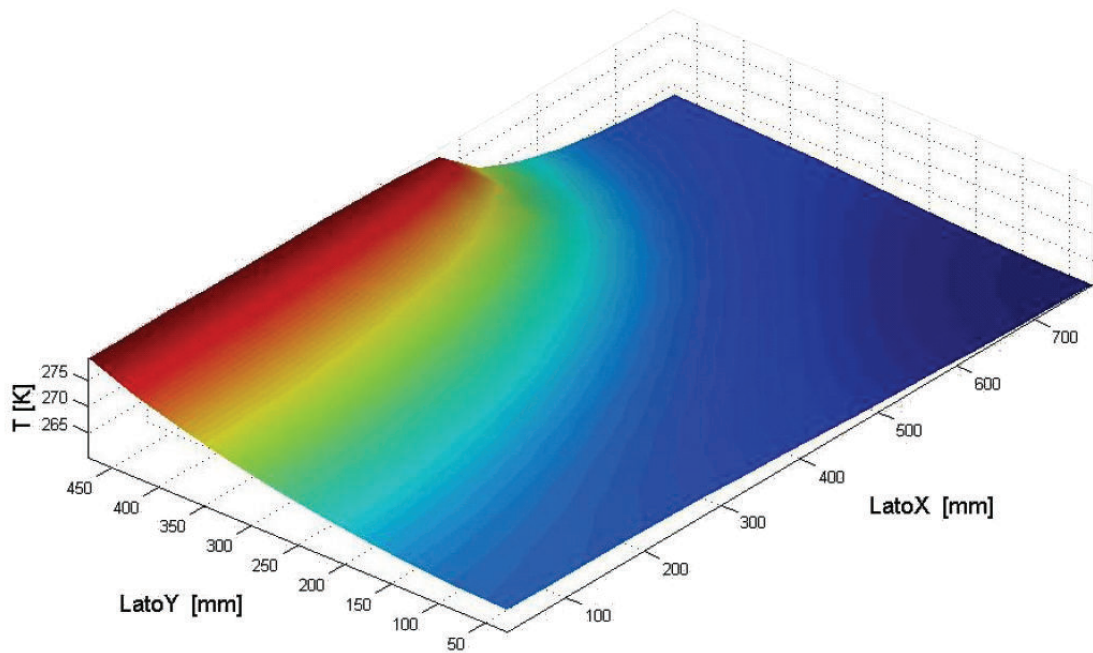


Figure 3.12: Example of a temperature distribution.

while the environment of the system was assumed to be at a given temperature (lower than the initial uniform condition of the panel). In this situation, the model has proved to be consistent in reaching the temperature of the surrounding environment, after a necessary period of transition. The results of an initial uniform panel temperature of 293 K and an outside temperature of 95 K are plotted in Figure 3.13, which shows the time evolution of the average temperature on the panel: the x axis reports the time (s) and the y axis reports the average temperature (K). For convenience, the graph only depicts the average temperature of the upper skin. It is possible to see from the figure how the results are compliant with previous observations, as can be seen from the value behavior that tends to the external environment temperature.

Simulations were also run to verify whether a physically correct time evolution of the average temperature in the presence of both a thermal load and a number of active nodes had been reached. An example of this kind of simulation is depicted in Figure 3.14, in which the time evolution of the average temperature of the upper skin is plotted in the presence of non-zero thermal loads. The x axis reports the time, measured in seconds, and the y axis reports the average temperature, which is measured in Kelvins. In this case, the value of the average temperature profile depends on the operative profile of the two electronic units.

One of the simulation concerned the activation of the two elements that represent the electronic loads, considering the time evolution. The two units were assumed to apply the same heat flux. No other heater was considered operative at the same time. The steady-state response was the one that is reported in Figure 3.15, which shows the temperature distribution on the upper surface with the electronic units activated. The [x y] plane contains the geometrical representation of the upper skin (the dimensions are in millimeters), while the z axis reports the temperature (in Kelvins). In this situation, the heat introduced by the electronics alone is sufficient

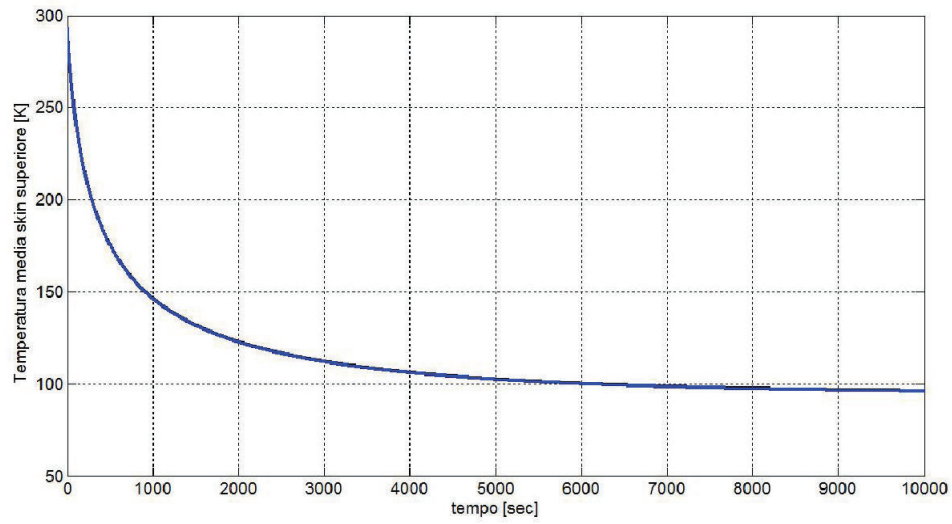


Figure 3.13: First validation case: time history of the average temperature of the upper skin.

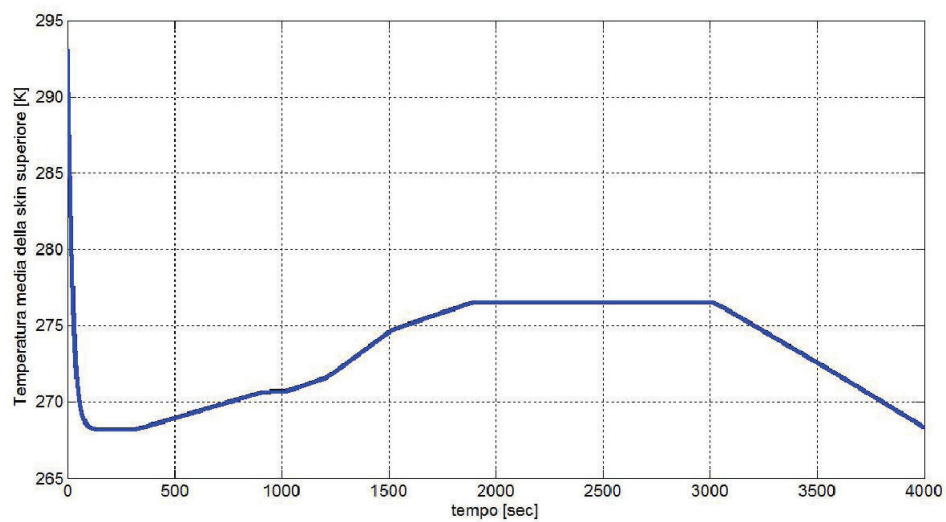


Figure 3.14: Second validation case: time evolution of temperature with non-zero thermal loads.

to balance the heat loss. This particular load configuration has been compared with experimental data from thermal vacuum tests and it has shown a good correlation.

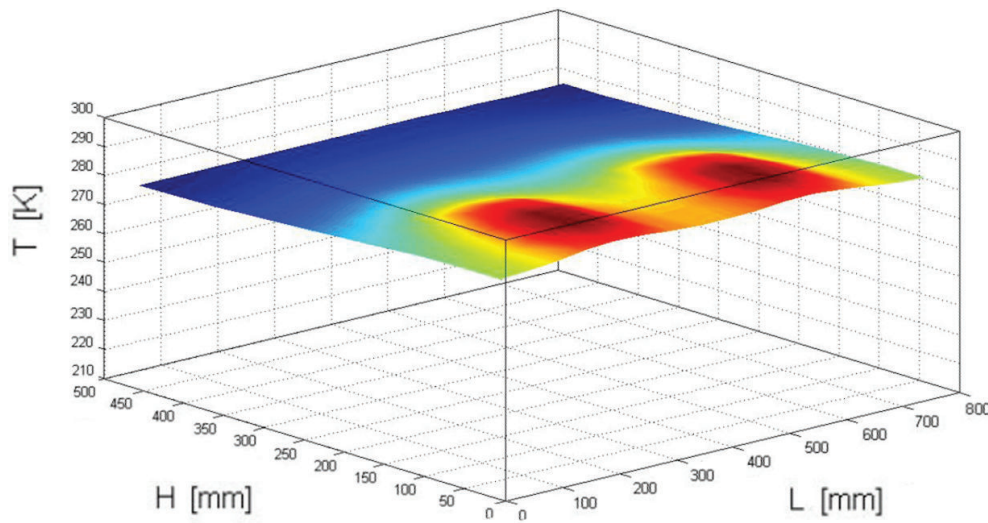


Figure 3.15: Temperature distribution on the upper surface with the electronic units activated.

After their construction and validation, the thermal and control models were integrated in a software environment that was capable of handling optimization in an automatic way: as in the previous optimization loops, Simulia iSight was chosen. Before the beginning of the optimization process, it was necessary to obtain preliminary knowledge on the influence of different parameters on the overall MFS performance (energy consumption and temperature uniformity). Initially, after the integration of the thermal management model with the design variables space, some Design of Experiments processes were conducted. The DOEs were mainly conceived to evaluate the reliability of the created model, but it was also possible to infer from them some useful considerations on the effects that could be obtained changing different variables. This phase was followed by the definition of the optimization study.

The optimization activity in itself was divided into three steps: the first optimization run allowed to get an idea of the best objective function values attainable if the heater layout was let free to change and their activation logic was chosen between two simple alternatives: ON÷OFF or proportional. Starting from the results of this step, two engineeringly feasible layouts were chosen, and a second optimization run was launched. This time other two control laws (PID control with fixed correction parameters, and PTC heaters) were added to the available set, and the algorithm was free to choose between them and change some of their parameters. Since the second step highlighted the superior performance of PID control, the third optimization run was aimed at improving the result by fine tuning of the parameters belonging to this control law.

With the first optimization studies it was possible to achieve a two-fold objective: to optimize the layout of the thermal control heaters in an improved ABB design, and

to evaluate the different solution methods available in iSight. The design variables considered in this preliminary phase were the layout of the nodes considered to host control heaters, the maximum power supplied to the control elements (i.e., the power supplied by the control element under a saturation condition), and the type of control logic (ON÷OFF or proportional). The solution algorithm was allowed to chose how many and which of the 160 potential control heaters would be activated. Moreover, the maximum power was allowed to range between a minimum value of 2 W and a maximum of 12 W. These limits reduced the calculation time, by restricting the analysis to values that are physically compatible with current technologies for space applications. Two objective functions were considered: the energy absorbed by the control equipment, and the evaluation of the temperature uniformity along the panel (in-plane distribution, upper skin). The value of the involved energy was computed at the end of a fixed simulation time. The purpose of this objective was to minimize the overall value obtained from the simulations. The goal for the index representing the uniformity of temperature along the panel, which was evaluated in the final condition (even though it is potentially possible to consider its values at any time point during simulation), was once again minimization. The assessment of temperature uniformity was obtained by analyzing the thermal gradient along the panel. In particular, the figure of merit is the average modulus of the bi-dimensional temperature distribution gradient:

$$\text{UnifTemp} = \frac{1}{N_n} \sum_{i=1}^{N_n} \|\nabla T_i\|$$

In the multi-objective situation, thanks to the definition of appropriate weight factors, it was possible to give more emphasis to the minimization of energy consumption, rather than to the maximization of temperature uniformity. As far as the constraints introduced into the system, on the basis of the above considerations are concerned, the three adopted quantities were the profiles of the operating electronics, the positioning of the electronic units, and compliance with the operative temperature range for the electronics. The definition of the operating profile for the electronics is reported in Figure 3.16 for this specific case. The representation shows the trends of the two quantities PE1 and PE2, which, for convenience, are represented by piecewise linear functions. The position of the units of the electronic groups is considered fixed on the panel.

Respect of the temperature requirements of the electronic units was also enforced: exiting from the operational temperature range was not allowed, even for a limited time interval. The maximum allowable temperature was 320 K, and the minimum temperature was 270 K. Evaluation of violations of this constraint condition was made by estimating the *ConVio1* and *ConVio2* variables for the two zones covered by electronics (respectively, left and right motherboards). *ConVio1* and *ConVio2* are computed by summing up the absolute value of constraint violation for each time step during the simulation, i.e. by evaluating the violation $|\Delta T|$ (for a single time step and for all electronic nodes) and then summing up all $|\Delta T|$ during the simulation:

$$|\Delta T|_{\text{time step } i} = \begin{cases} T_{\text{MIN}} - T_{\text{ele}} & \text{when } T_{\text{ele}} < T_{\text{MIN}} \\ T_{\text{ele}} - T_{\text{MAX}} & \text{when } T_{\text{ele}} > T_{\text{MAX}} \end{cases}$$

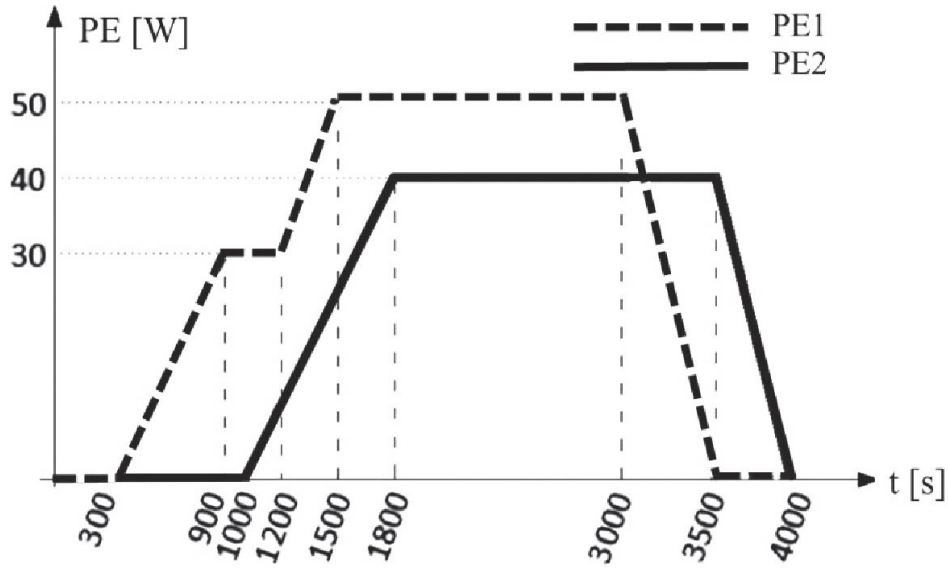


Figure 3.16: Operating profile of the electronics units.

$$\text{ConVio} = \sum_i |\Delta T|_{\text{time step } i}$$

where T_{MAX} and T_{MIN} are upper and lower operative temperature limits, and T_{ele} is the generic temperature T_i of node i belonging to one of the two electronic motherboards (subscript 1 or 2 has been omitted for ease of reading). In addition to the aforementioned constraints, the temperature of the boundary nodes that are interfaced with the bottom skin elements was set uniformly equal to 95 K and kept constant over time. This definition reflects the boundary conditions applied during laboratory tests in a thermal-vacuum chamber.

Six different optimization cycles were run with these variables, constraints and objectives, each of which was based on a different solution algorithm: Archive-Based Micro Genetic Algorithm (AMGA), Adaptive Simulated Annealing (ASA), Evolutionary Optimization Algorithm (Evol), Multi-Island Genetic Algorithm (Multi-Island GA), Neighborhood Cultivation Genetic Algorithm (NCGA), and Non-dominated Sorting Genetic Algorithm (NSGA-II). The research used an available set of optimization tools: the selected algorithms all belong to the evolutionary family, because this category is more suitable for non-linear functions problems in which the design space includes discontinuous variables. The time required by the six algorithms to resolve the problem was very similar (~ 17 hours, except for ASA, 19.5 hours, and Evol, 18 hours, on a Quad-Core, 3 GHz, 12 MB cache processor).

The real ABB panel hosts the equivalent of 38 active (heater) nodes, and their distribution is shown in Figure 3.17, while the different layouts obtained with the optimization studies are reported in Figures 3.18 to 3.23. In these figures, the dark grey squares represent the footprints of the motherboards, while the light grey squares indicate the presence of heaters. The numerical optimization results are summarized in Table 3.17. Solving the problem with six different optimization algorithms leads to six different optimal solutions. Therefore, for each one of the aforementioned optimization algorithms, the table presents a description of search

settings and results. The first column on the left specifies the names of the algorithms. Then a second column contains the main configuration parameters of the algorithms themselves (being evolutionary algorithms, their search behavior is dictated by these settings). Finally, the next five columns present the characteristics of the optimal solution found by each algorithm. Three columns out of five describe the value assigned automatically by the optimization algorithm to the free parameters: control logic (ON÷OFF vs. proportional), number of active heaters (0÷142), and maximum power that can be dissipated on each single heater patch (2÷12 W). The last two columns on the right give the value of the objective functions for the optimal solution: total energy required to keep the structure inside the operative temperature range during the whole simulation, and temperature uniformity index. As a basis for comparison, the actual ABB configuration, with the ON÷OFF control logic, requires 0.0576 kWh of energy and results in a temperature uniformity index equal to 2.02.

Table 3.17: Results of heater layout optimizations.

Name of Algorithm	Configuration parameters	Parameters and figures of merit of the optimum solution				
		Control logic	Active heaters	QHmax (W)	Energy (kWh)	T Uniformity Index
AMGA	Archive Size Limit = 100 Crossover Distribution Index = 10.0 Crossover Probability = 0.9 Initialization Mode = Random Population Size = 80 Mutation Probability = 0.5 Number of Function Evaluations = 1000	Proportional	43	5.733	0.0380	1.56
ASA	Convergence Epsilon = 1.0E-8 Initial Temperature of the Parameters = 1.0 Max. Number of Failed Designs = 100 Max Number of Generated Designs = 1000 Min. Ratio of Accept. Designs for Re-annealing = 1.0E-6 Number of Designs for the Convergence Check = 5 Number of Accept. Designs Before Re-annealing = 100	Proportional	25	11.329	0.0385	1.61
Evol	Max. evaluations = 1100 Convergence tolerance = 0,01	Proportional	47	9.163	0.0423	1.40

Table 3.17: Results of heater layout optimizations.

Name of Algorithm	Configuration parameters	Parameters and figures of merit of the optimum solution				
		Control logic	Active heaters	QHmax (W)	Energy (kWh)	T Uniformity Index
Multi-island GA	Interval of Migration = 5 Number of Generations = 10 Number of Islands = 10 Rate of Migration = 0.01 Rate of Mutation = 0.01 Sub-Population Size = 10	Proportional	41	8.369	0.0407	1.47
NCGA	Mutation Rate = 0.01 Number of Generations = 50 Population Size = 20 Crossover Type = 1 (single point crossover)	Proportional	35	5.183	0.0348	1.55
NSGA-II	Crossover Probability = 0.9 Initialization Mode = Random Number of Generations = 50 Population Size = 20	Proportional	37	5.980	0.0370	1.22

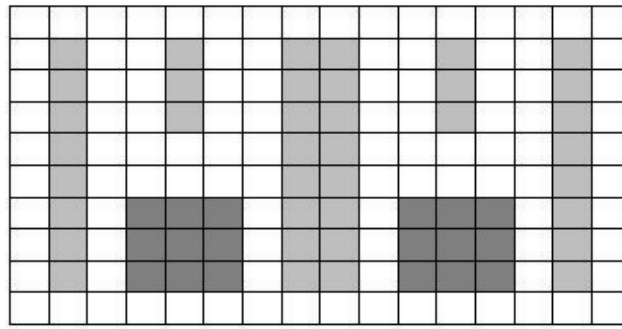


Figure 3.17: Heater layout on ABB during the thermal vacuum test.

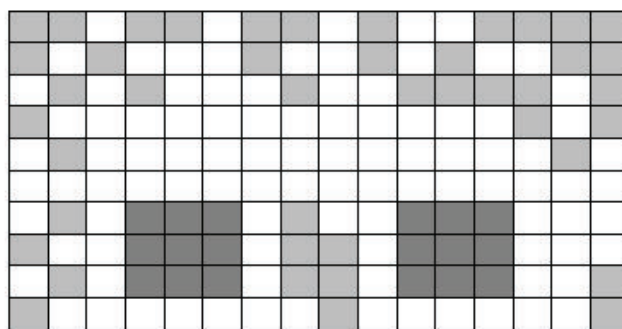


Figure 3.18: AMGA Optimal heater layout.

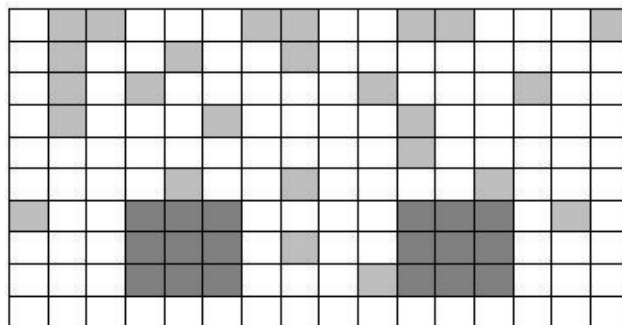


Figure 3.19: ASA Optimal heater layout.

Summing up, the combined effect of an optimized heater layout and a proportional control logic led to a saving in energy consumption between 26.6% (Evol) and 39.6% (NCGA), and an improvement in temperature uniformity of 20.3% (ASA) to 39.6% (NSGA-II). Another good outcome of this study is the fact that almost all the different algorithms identified the same areas that needed to be controlled via heaters (i.e., opposite corners of the top half of the panel and borders around the electronic motherboards). This fact both confirms the reliability of the computed solutions and offers a useful indication to select a finite number of layouts. However, the geometrical arrangements of the heaters found thanks to the optimization are not readily implementable with standard thermal hardware; being too scattered, the heater patches would create a problem in terms of wire

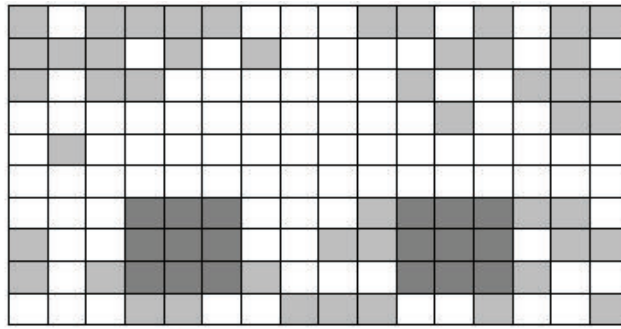


Figure 3.20: Evol Optimal heater layout.

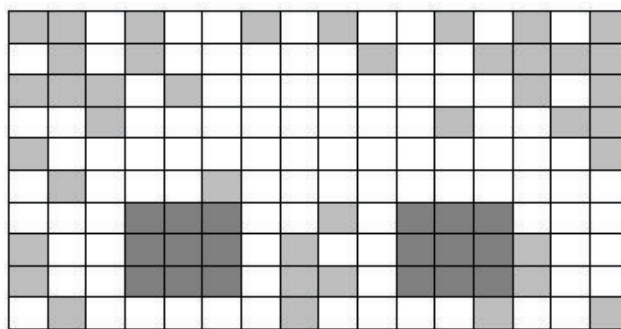


Figure 3.21: Multi-Island GA Optimal heater layout.

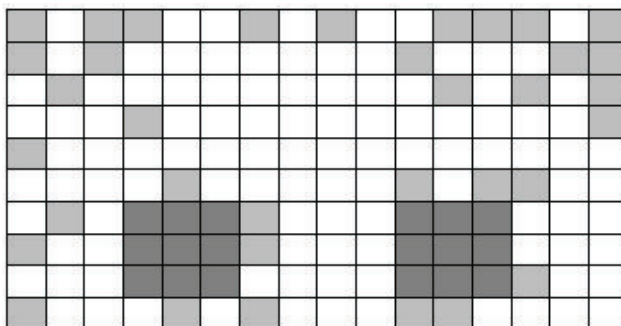


Figure 3.22: NCGA Optimal heater layout.

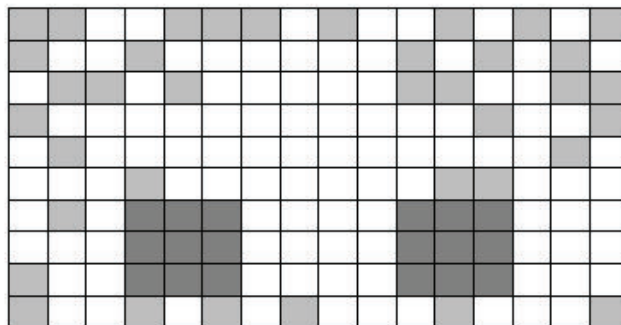


Figure 3.23: NSGA-II Optimal heater layout.

harness congestion. The benefits foreseen with such a layout have motivated further investigation and have given rise to a new concept for the smart thermal control layer of multifunctional systems, which would allow the exact heater distribution resulting from optimization to be reproduced. This new concept, with a distributed heater architecture, is currently under study in the framework of the ROV-E project in order to solve issues related to cable length, routing, and connection to the power distribution unit. The remainder of this study will only deal with layouts that could be implemented with already existing and proven hardware.

At the present time, with a few exceptions, thermal control heaters on board satellites are piloted with an ON÷OFF tactic. Even though this approach offers some undeniable advantages, in terms of simplicity and reliability, it is clearly not optimal for energy saving policies or for applications with strict temperature stability requirements. Therefore, to encourage migration towards more energy efficient control strategies, three additional logics were selected and compared, in order to identify the benefits and drawbacks: although the ON÷OFF logic was maintained, a proportional scheme, a PID (Proportional, Integral, Derivative) controller, and PTC (Positive Temperature Coefficient) heaters were added. As reported in Figure 3.24, the ON÷OFF control is centered on a control temperature (T_{ctrl}) and has a sharp hysteresis loop (between $T_{ctrl} - 5\text{ K}$ and $T_{ctrl} + 5\text{ K}$).

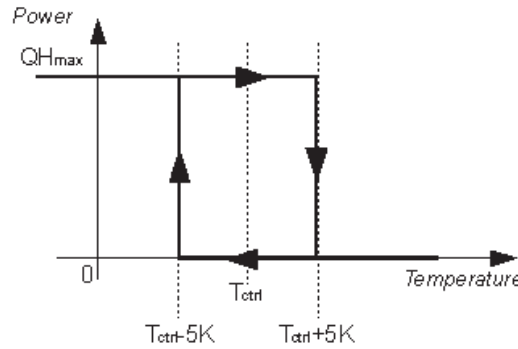


Figure 3.24: ON÷OFF control (with hysteresis).

The proportional rule is also quite simple. As shown in Figure 3.25, it is based on two reference temperatures, a limit one (T_{lim}) and a threshold one (T_{th}): when the node temperature is lower than the limit value, the heater is turned on at maximum power, on the other hand, when the node temperature is higher than the threshold value, the heater is turned off. Linear behavior, which decreases the dissipated power from Q_{Hmax} to 0, lies in-between.

It is not worth describing the PID controller in detail, since this control mechanism, which is based on a feedback loop, is well known and used extensively in many engineering fields. The controller is assumed in its ideal parallel form, in which the proportional, integral and derivative contributions have three different and independent coefficients:

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{d}{dt} e(t)$$

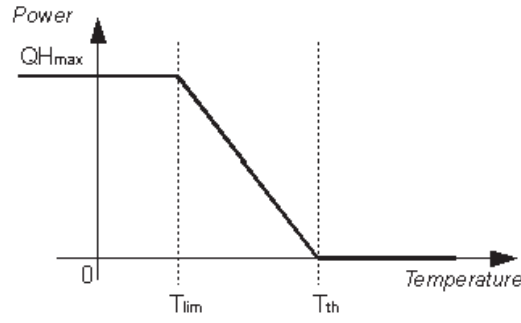


Figure 3.25: Proportional control law.

where $u(t)$ is the command signal, $e(t)$ is the error signal and K_p , K_i , and K_d are the tuning gains. The three multiplicative constants are the same for each heater on the panel. A fixed set point temperature (T_{sp}) is used to compute the error signal and is compared with the node temperature. A saturation law for $u(t)$ has also been added, in order to represent the physical limitation of the maximum power that can be dissipated on a single heater.

PTC heaters are resistors that exhibit a slightly negative temperature-coefficient in the lower part of their temperature range, a sharp rise in resistance around the Curie point, and a negative coefficient again at higher temperatures. As can be seen in Figure 3.26, their $R(T)$ curve is like a steep sigmoid and it is theoretically defined by four characteristic parameters: the zero power resistance at 25°C (the resistance value measured at 25 °C in such conditions that the change in resistance due to the internal generation of heat is negligible, $[R_{25}]=\Omega$), the minimum resistance (the minimum zero power resistance of the component, measured at the temperature in which the coefficient becomes positive, $[R_{min}]=\Omega$), the switching temperature (at which the resistance is twice the minimum resistance, $[T_b]=K$), and the temperature coefficient of resistance (the value of the slope of the resistance-temperature curve in the increasing resistance range, $[\alpha]=\%/K$).

$$\alpha = \frac{100}{(T - T_b)} \ln\left(\frac{R}{R_b}\right)$$

Since PTC components can be tailored to the user's needs, a fixed $R(T)$ curve has been assumed and T_b has been kept as a parameter (indicated as T_{rise}).

In conclusion, a block model was built for each of the four aforementioned logics. The blocks are all gathered in the same subsystem, and the Simulink model is able to switch between the possible controllers by operating a four-port switch.

As already seen, a certain number of possible heater configurations was obtained from the first optimization study. These layouts were obviously not immediately applicable as engineering solutions, since they were too granular. Therefore, a couple of realistic arrangements have been taken from those scattered maps that include a total of 25 to 47 heaters: 29 active nodes were distributed in a compatible manner with commercially available rectangular Kapton or tape heaters. A degradation in panel isothermia is very likely to be introduced into the problem with this set up; however, these adjustments are necessary to obtain a viable design.

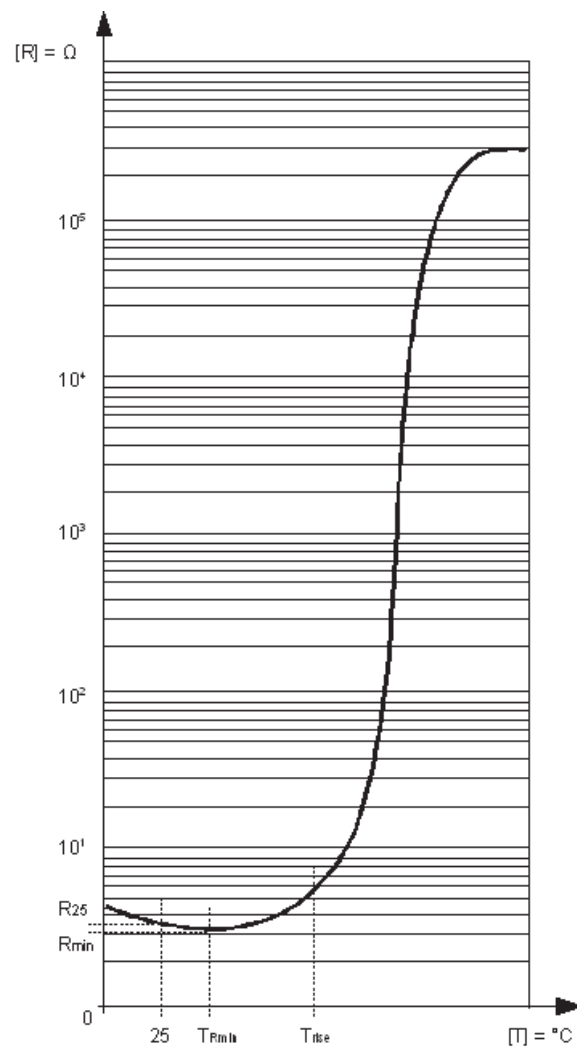


Figure 3.26: $R(T)$ curve typical of PTC heaters.

Once the distribution of the active nodes has been fixed on the upper sandwich skin (as shown in Figures 3.27 and 3.28), a second optimization activity was started, this time aiming at the selection of the thermal control law and adjustment of its parameters. The definition of the problem is reported in Tables 3.18 and 3.19. The problem has been solved with an NCGA algorithm, based on a population size equal to 10, a number of generations equal to 20, single point crossover with a crossover rate equal to 1.0, and a mutation rate equal to 0.01.

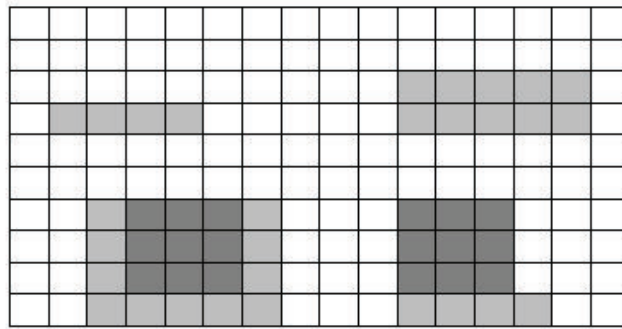


Figure 3.27: Heater layout number 1.

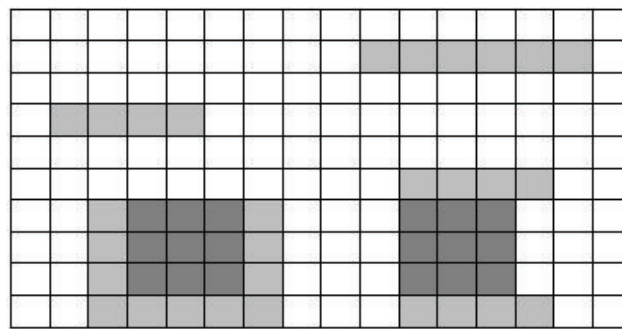


Figure 3.28: Heater layout number 2.

The optimization process identified an optimal solution, characterized by a PID control law with $T_{sp} = 275$ K, applied to heater layout number 2. The maximum power for each heater is 4.08 W. The energy consumption with this configuration is about 0.0303 kWh, while the temperature uniformity index is 2.21. A reduction of ~ 1 W on the smallest allocated $Q_{H_{max}}$ and a total energy saving ranging from 4.5 to 12 Wh were obtained with respect to the solutions of the first optimization study. On the other hand, a relevant increase in the temperature uniformity index was measured, reflecting a worse solution from the isothermality point of view. However, this is mainly due to the correction that has been applied to the heater layout in order to make it feasible from the engineering point of view.

The optimum points of the first optimization run were obtained using a proportional logic, while the best solution identified in this optimization referred to a PID controller whose coefficients have been determined with the Ziegler-Nichols method ($K_p=8.4$, $K_i=67.2$, $K_d=0.1$). Table 3.20 shows a numerical comparison. The columns labeled as “Proportional” summarize the results of the old optimum point, i.e. the optimal solution of the study presented before. The first row contains