THE CRONOS CODE

General structure
CRONOS is an environment dedicated to integrated simulation. It is organised into a suite of computer codes and a graphic interface. The code is modular and “open source”. Users choose the modules they want to use and the periods for which they want to evaluate them. The modules are called “self-declaring” in the sense that they provide the information required for the dynamic creation of the parameters specific to them and the associated graphic interface. The main body of the programme and the graphic interface have both been developed in MATLAB. However, many of the modules have been written in FORTRAN (which represents the majority of the lines of code) and a few in C or C++. Some modules use the NAG library.

The modules are linked to MATLAB via an interface called “mexfile”. The MATLAB part of the code uses the “signal” toolbox. The use of the advanced functions, mainly for controlling the tokamaks and the controls also requires the use of the SIMULINK tool.

The CRONOS data set consists of MATLAB data structures. The standard data set consists of 3 structures:

- param: contains all the non time-dependent data
- data: contains all the time-dependent data
- post: contains all the data generated outside the execution of CRONOS

The CRONOS input data set has exactly the same structure as the output data set. Only the values change. The time base of the input file sets the times for which the input data changes. The time base for saving data is specified by the same time base. The CRONOS data is IEEE double precision real values (64-bit), except for certain 2D data connected with equilibrium. Each user can add any necessary data to it and dynamically regenerate the code which handles the data.

Figure 1: CRONOS software environment (use of the GSL library is planned instead of or as well as the NAG library, the MATLAB Java API could be useful for accessing some databases or creating a Web interface for CRONOS)

CRONOS (figures 1 and 2) is available for alpha processor machines running under “OSF true 64”, on X86 (and shortly X86-64) machines under Linux (PC, JAC and PC farm).
For the compilation, CRONOS requires a development environment: MATLAB, NAG, LAPACK, BLAS, C and C++, FORTRAN (77 & 90) compiler. Under OSF, the FORTRAN used is the native FORTRAN of the machine. Under Linux, we use the “Portland” compilers (including the FORTRAN compiler).

CRONOS is generally used via the graphic interface. This enables the user to perform the following:
1. Prepare the data sets, either by connecting to the databases (TS, JET, FTU, TCV, ITPA DIII-D international database), or by preparing a data set from scratch (useful for future machines such as ITER or DEMO, or machines for which direct access to the data is not possible), or based on a CRONOS file corresponding to a previous simulation.
2. Edit and modify the parameters and data in a data set. It is also possible to modify the parameters in command lines without using the interface (including the SIMULINK functional diagrams).
3. Carry out an interactive or “batch” run of a CRONOS “case”.
4. Display the data using pre-provided functions, a generic display tool (zdataplot) or creating one’s own diagrams (with MATLAB language graphic tools).
5. Manipulate the data sets (modification of the version, the time base, the number of radial points, data import/export, manipulation of SIMULINK functional diagrams)
6. Launch post-processes for calculating data as produced by the diagnostics.
7. Launch “wizards”. The wizards are fast computer codes used to check the coherence of the data, adjust the modules or develop scenarios.
8. Manage the access paths and generate the dynamic code and online help.
9. Certify a CRONOS version (partly functional: verification of the syntax of “mfiles”, detection of any change of source file, testing of standard cases and comparison of the results).

It is also possible to handle the data directly in the MATLAB workspace.

The CRONOS core is modular, in the sense that the source modules, the transport coefficients, and the equilibrium, MHD stability, and neoclassical calculation modules, are defined at graphic interface level and dynamically linked to the CRONOS computer code (see figure 3). Each user can add modules directly (placing the modules in his environment) or modify any CRONOS function or module.
The computer code is based around a core in MATLAB. This core loads the data, carries out intermediate and final saves and manages the rerun of “cases”, writing the trace file and carrying out the calculations. These start with the data initialisation, then the code enters the time loop. The internal time step is calculated automatically in CRONOS. The input data is defined on a time base built by the user. Data such as the settings are linearly interpolated between two input times when the time intervals are split into slices. CRONOS manages the events (which are defined as being changes of state of the plasma not solved temporally in CRONOS, such as the injection of pellets, MHD reconnections, ELMs). In the time splitting module there is a function which performs the solution for a basic time interval. In this function, the electron and ion heat diffusion/convection, electron density, toroidal rotation and current diffusion equations are solved, and the ordered call to external modules (equilibrium, neoclassical, wall, transport of impurities, sources, MHD stability) is performed.

The diffusion/convection equations are currently solved using a mixed implicit/explicit “solver” included in a convergence loop on the non-linearities of the transport coefficients. The modules for calculating the transport coefficients are called in this loop. In the standard operating mode, the call to the neoclassical module and the calculation of the radial electrical field are also included in this loop. It is also possible to include in this loop the call to the source modules for the description of certain fast transients. The call to the equilibrium is asynchronous in relation to the time splitting.
CRONOS modules
A large number of modules are implemented in CRONOS and it is very easy to add more. These modules address the various problems associated with the physics of tokamak plasmas:

1) Magnetic equilibrium and MHD
   a. HELENA code (2D magnetic equilibrium)
   b. MISHKA and CASTOR codes (linear MHD stability)
   c. Reconnection modules for sawteeth
   d. KINEZERO code (linear stability, gyrokinetics)

Heat sources (matter, current, rotation)

   e. PION code (ion cyclotron waves, minority + harmonic heating)
   f. ABSOR code (ion cyclotron waves, FWEH)
g. REMA code (electron cyclotron waves)
h. SINBAD code (neutral injection)
i. DELPHINE and LUKE codes (Lower Hybrid waves, including a 3D Fokker-Planck module)
j. SPOT code (Monte-Carlo code describing the distribution of the alpha particles produced by fusion reactions)
k. CYTRAN and EXATEC codes (losses of energy by synchrotron radiation)

2) Sources of matter
   a. SOL-ONE and JONASSLICE codes (sources of matter at the plasma edge)
   b. GLAQUELC code (Deposition models for pellets)

3) Neoclassical theory (bootstrap, rotation)
   a. NCLASS (neoclassical transport coefficients)
   b. Sauter model (bootstrap and resistivity)

4) Impurities
   a. ITC code (transport of impurities)

5) Transport models (non-exhaustive list)
   a. zbgbs_ts : Bohm/Gyro-Bohm, (optimised for Tore Supra)
   b. zbgbs : Bohm/Gyro-Bohm, (optimised for JET)
   c. zbgbs_rot : Bohm/Gyro-Bohm, (optimised for JET, with rotation effects)
   d. zweiland : Weiland model
   e. zglf23 : GLF23 model
   f. zforcefree : so-called “force-free” model
   g. zetg_stable, zitg : Horton critical gradient model (ETG and ITG modes)
   h. zkiauto : transport model based on 0D energy confinement time scaling laws.

**Pre/post processing and display tools**

The pre-processing modules of CRONOS concern essentially access to the various databases with the TSLib and MDS+ protocols, in order to fill the CRONOS structure. For the moment the following tokamaks are accessible in CRONOS: Tore Supra, JET, and FTU. Coupling to the Tore Supra database is achieved by means of the TPROF code with regard to density and temperature profile fits.

For the other tokamaks, access to the various databases is achieved in a transparent fashion via MDS+. A density and temperature profile fit procedure has also been written, applicable for any fit involving local data not enabling the use of integrated data on a line of sight (JET, FTU, TCV, DIII-D). This procedure is automated (very few adjustments necessary).

The post-processing modules concern the reconstruction and/or computation of data using the data created by a CRONOS simulation. The majority of the post processing modules address the reconstruction of diagnosis signals in order to validate a simulation. The following are thus reconstructed:
- Faraday angles (validation of the current profile)
- MSE angles (validation of the current profile)
- line-averaged densities
- ECE temperature radii

Another post-processing module concerns the linear stability of the plasma vis-à-vis MHD phenomena. The MISHKA or CASTOR codes can be initiated at several times in order
to look at the growth rate of the various MHD modes. These data are saved in the “post” structure of CRONOS.

It is possible, through the interface, to retrieve the simulation outputs at a given time and to separately re-launch the modules that were or were not used in the simulation, having the possibility of modifying the parameters of these modules. In this case the result will not be saved in the result file. For the moment the following modules are accessible:

- PION (ion cyclotron wave deposition module, study of the effect of the concentration of minority ions and the position of the cyclotron layer)
- REMA (electron cyclotron wave deposition module, the effect of the poloidal and toroidal injection angles)
- The electron density profile (method of inverting the linear density by maximum entropy with or without local constraints; reflectometry)
- KINEZERO (linear stability module of the ITG/ETG/TEM modes)

A rapid simulation programme (zero-dimensional) can be launched before or after any complete simulation of CRONOS (see next section). It allows comparison with scaling laws.

Finally, graphical tools have been developed in order to display a simulation of CRONOS:

- zdataplot function (for tracing all the CRONOS data, as a function of time and space)
- Coherence of a simulation for Tore Supra (comparison with the experimental signals)
- Comparison of the various bootstrap models
- Comparison with the 0D scaling laws

Once a result file of a CRONOS simulation has been loaded, it is very easy under MATLAB to make your own display scripts, all the data being accessible.

Verifications and validations

The equations solved in the CRONOS code (diffusion / convection of current, heat, matter and rotation) have all been validated both from the theoretical point of view (with a mathematical demonstration of these formulae) and from the practical point of view (by application to simple cases). The current diffusion equation

\[
\frac{\partial \Psi}{\partial t} = \frac{\left< \nabla \rho^2 \right>}{\mu_0 \sigma_{\parallel} \rho_m^2 \left( \frac{1}{R^2} \right)} \frac{\partial^2 \Psi}{\partial x^2} + \frac{\left< \nabla \rho^2 \right>}{\mu_0 \sigma_{\parallel} \rho_m^2 \left( \frac{1}{R^2} \right)} \frac{\partial}{\partial x} \left[ \ln \left( \frac{V^2 \left< \nabla \rho^2 \right>}{R^2 F} \right) \right] + \frac{x}{\rho_m dt} \frac{d \rho_m}{\partial x}
\]

\[
+ \frac{B_0}{\sigma_{\parallel} F \left( \frac{1}{R^2} \right)} j_{ni}
\]
was entirely re-derived from fundamental equations of plasma physics (generalised Ohms law). The method (Crank-Nicholson) of solving this equation has also been tested on cases where the solution is analytical.

The terms deduced from the 2D magnetic equilibrium \( \left( \frac{V \rho}{R^2}, \left( \frac{1}{R^2} \right), V' \right) \) have, in the case of a circular equilibrium, been validated with respect to another equilibrium code, VMOMS (valid for circular cases) and with respect to analytical formulae.

Moreover, the various modules have undergone external validation by their authors, before being integrated into CRONOS.

Verification of the data is an essential point, since simulations may lead to non-physical results. It must be applied during the preparation of the data, and to the results of the simulations as well.

During the preparation of the data, it is important to verify the minimum coherence of the starting data in order to ensure a maximum chance of success with the simulation. This concerns both the sets of data downloaded from the databases and the simulations prepared from scratch. For simulations prepared from the databases of existing machines, this function is provided by the processing operations and the data fit tools (TPROF for Tore Supra, ZJET for JET, ZFTU for FTU, etc.). For all simulations, it is possible to verify a certain number of parameters using the 0D tool associated with CRONOS. Finally, the verification of the data is also based on the expertise of the user and the direct display of the data.

After simulation, it is imperative to verify its result. This is done firstly by displaying the data. A certain number of simple verifications can be made by comparing the results with the measurements, for simulations of existing discharges. For Tore Supra data, there exists a specific assistant that automatically traces these data (TS consistency).

The 0D (0 dimension) assistant associated with CRONOS makes it possible to check that the CRONOS simulation gives results compatible with the 0D simulation (the 2D and 1D data of CRONOS make it possible to calculate all the 0D data simply). The 0D tool of CRONOS (called METIS) makes it possible to prepare the scenarios of the simulations and to quickly verify the results of the simulations (the calculation time is around one minute). It uses scaling laws (confinement times, current drive efficiencies, profiles of peak factors, radiated powers, \( Z_{\text{eff}} \)). It integrates simple differential equations in order to resolve the time evolution of various quantities. It calculates all the current sources (bootstrap, LH, ECCD, NBCI, FWCD and fusion alpha). It calculates the energy content of the plasma, including the supra-thermal contributions, as well as the distribution of the power between the ions and electrons. It supplies a prediction for the temperatures and the mean densities. It estimates the power sources (LH, ECRH, FWEH, ICRH, NBI, fusion) and radiated powers (bremstrahlung, line radiation, cyclotron radiation). It precisely calculates the fusion power (using the formulae of the effective cross-section, passing to the profiles again, and then re-integrating). It takes account of the retention of helium in order to calculate the change in the \( Z_{\text{eff}} \). It contains a convergence loop on the non-linearities (in particular that associated with the fusion power). It also contains a current diffusion pseudo-equation that uses an estimation of the position of the current sources and that makes it possible to predict the change in internal inductance \( I_1 \) and the loop voltage \( V_{\text{loop}} \).

Alongside these tools, CRONOS has assistants making it possible to adjust the modules of the sources (REMA, PION and SINBAD), as well as assistants for using the CRONOS data to compute the output of the measurements as would have been produced by the diagnostics (interferometry and reflectrometry, ECE via REMA, etc), for a given time.
Figure 4: Change in the Faraday angles during the TS 28205 shock. Circles: values measured by the polarimetry; solid lines: Cronos prediction.

Figure 5: Change in the quantities relating to the current profile during the JET 53521 shock. The measured values are compared with the CRONOS prediction, which use DELPHINE for calculating the current generation by LHCD waves, and the result of TRANSP for the injection of neutrals.

The use of all these tools makes it possible to compare the CRONOS predictions with the experiments and in particular to validate the code in current diffusion mode. On Tore Supra, the hard X-ray tomography diagnostic is often used for deducing the source of LH current, which provides reliable reconstructions of the current profile. It is then possible to compare the result obtained with the measurements available: internal inductance, MHD events (sawteeth, tearing modes, etc), polarimetry (see Fig. 4). Coherent integration of this information is used to validate the current profiles predicted by CRONOS, as well as the modules used for estimating the current sources on Tore Supra.

On JET, the same type of comparison is possible and has been used to validate both the diffusion of the current and the modules of the current sources used (Fig. 5).
The correct implementation of the models in the transport codes can be verified by comparing the results obtained with two distinct codes. It has been possible to make isolated comparisons with JETTO, concerning the solution of the basic equations as the diffusion of the current. Good agreement between the predictions of the two codes has been found in ohmic plasmas. It would be desirable in future to make this type of comparison more often and more systematically, in order to increase the probability of identifying errors in the codes. This is nevertheless an arduous and somewhat thankless task, which requires the mastery of several transport codes, and few volunteers are found for the moment. Recently, an attempt has been made to compare the predictions of sophisticated transport models such as GLF23 or the Weiland model, implemented in JETTO, CRONOS, ASTRA and XPTOR. Differences are often found between the predictions, the resolution of which would require an effort made by several experts dispersed in different laboratories. Validation of the implementation of the models in the codes and the code benchmark remain to be developed in the integrated modelling community.

**Coupling with DINA-Simulink**

Simulink is an extension of MATLAB dedicated to the simulation of the dynamic systems. It is based on a graphical interface that makes it possible to draw graphically the dynamic systems by means of the base element (integrator type, filter, PID, multiplier, adder, etc). It is particularly suited for simulation of signal processing and control systems. It also includes a state machine (for simulating, for example, a programmable automatic controller).

As from version 3.0 of CRONOS, it is possible to use “block diagrams” of SIMULINK for specialised slavings and/or calling of S-Function. CRONOS is then seen by SIMULINK as an S-Function connected in a “block diagram”. SIMULINK supplies the temporal sequencing to CRONOS, which is synchronised on top. The “block diagram” is then feedback capable of modifying the requests used by CRONOS, but also data of the profile type. All the CRONOS data are available, on request, in SIMULINK. To this end, it is necessary to write a gateway function for each class of problem. This is done from an example that is simple to modify. This constraint is due to the current structure of SIMULINK.

The CRONOS interface makes it possible to directly manipulate the “block diagrams” and to set their parameters, to launch SIMULINK and to set its execution parameters as well.

The coupling of DINA (equilibrium code with free separatrix taking into account the chamber and the poloidal system, able to solve the heat, density and current diffusion equations) and CRONOS then consists of drawing a “block diagram” involving the DINA S-function and one of the CRONOS S-Functions. This coupling works for the moment in a simple manner (the separatrix is passed from DINA to CRONOS and various fields, including the sources, from CRONOS to DINA). Provision is made for extending it and for making a version of DINA able to replace the CRONOS equilibrium.
Feedbacks

The feedback modules are naturally implemented in CRONOS. These modules allow the simulation of various types of feedback (proportional, integral, etc), to calculate in each case the gains and the transfer matrices. They allow, by means of a memory structure, the recovery of the values of the actuators as well as their control (power, density, etc requests).

Here is a list of the slaving modules already set up in CRONOS (setting parameters of gains, etc, by virtue of the interface):
- feedback on the plasma current with $V_{\text{loop}} = 0$
- feedback on the LH power with $V_{\text{loop}} = 0$ and given plasma current
- feedback on $\beta_{\text{pol}}$ with the LH power and ICRF
- feedback on the $q$ profile with the LH and ICRF power
- feedback on the fusion power with the LH power and the injection of matter

An important application of CRONOS with regard to the feedbacks has been the development of the “Extreme Shape Controller” for JET.

Interface with the databases

The interfaces with the databases serve to translate the experimental data of each tokamak, which are stored in different formats and different access modes, into variables of the CRONOS data structure, identical for all machines. Such interfaces in general require the setting up of pre-processing software, which will construct profiles that can be used by CRONOS, from the data of the diagnostics (TPROF on TS, ZJET on JET, PREJET/MODEX...
in JAMS (JET), ZFTU on FTU, etc.). It is also necessary to set the CRONOS modules according to the specific elements of the machine (for example position and characteristics of the heating antennas, diagnostics, etc). The only exception to these rules is the case of the ITPA database, which already contains directly usable profiles and pre-calculated power depositions.

From a technical point of view, MATLAB explores the databases via Tslib (TS) or MDS+ for the other cases. MDS+ is today very widespread in the fusion community, and it is essential to have the MDS+ functionalities in MATLAB. There exist communication protocols/structures for the databases that are more widespread in the world of data processing (DXML, HDF5, etc), the use of which is envisaged for future activities of the European Integrated Tokamak Modelling Task Force. However, MDS+ seems to meet the requirements of the fusion community in the short and medium terms.

CRONOS is today coupled with the databases of TORE SUPRA, JET, FTU, TCV, DIII-D, and that of the ITPA, and can set up an input file from these data. In the case of JET and TORE SUPRA, it is also possible to write the results of a CRONOS simulation in the database, in a form accessible even to persons not at all knowing how to use CRONOS.