The project was divided into two parts, thus there are two main goals.

1 Modeling of the plant

The first part of the project was devoted to the development of the dynamic model of the engine, using a high-level programming environment (namely, Mathworks MATLAB®). This presents a problem, because of a very high complexity of the device. We used as a prototype a type NPT171 engine with a 80 kg thrust and 15 gram per second mean fuel consumption. This motor is presently used with the ultra-light, pilotless airplanes and is capable of operating on heights of up to 30000 ft, sustaining speeds of up to 0.9 mach (sub-sonic speeds).

The preliminary analysis of the engine was conducted by our instructor, Dr. M. Lichtsinder. He provided us with the initial set of equations (27 relations that describe the static case) and translation maps (not all parameter relations within the engine can be described in the closed form mode; some of the parameters need to be experimentally evaluated and recorded in the special table, that is called map). By manual analysis it became possible to rearrange a mathematical description in a way to produce a complex sub-defined equation set, counting five equations and six free variables (the development of dynamic model raised these numbers to six and seven respectively). It was proved possible to obtain a complete engine parameter set from these six (seven) parameters after a solution of equation set.

The solution was found to be extremely problematic, because of complex dynamics of the model. Inability of Gauss-Newton type solver to converge to the solution, forced us to seek another way to solve a problem. After trying different solver types (between them Genetic Algorithm solvers, that were found inadequate due to extreme computational requirements), we developed a way to find a
solution. This was done with an aid of two primary methods. The first was to extrap-
olate the translation maps by means of the punishment functions. The second 
was to use two level Gauss-Newton solver, where low-precision, large-step solver 
was used to guide a high-precision, small-step solver into the regions of possi-
bile convergence. Even than, computational requirements were enormous, so we 
limited ourselves to the solutions along few trajectories, that were assumed to be 
typical. The common case solution will require a development of high-speed, op-
timized solver software and parallel computation facility (probably system of the 
SGI Origin 2000 class to achieve a near real-time ability). After the completion of 
the computations, the obtained data was compared with an available experimental 
data and found sufficiently compatible.

2 Model analysis and control

The second goal of the project was the utilization of the developed model in the de-
sign and evaluation of the speed control mechanism for the given engine. Because 
there is no known method of controller development for the arbitrary non-linear 
plant (except for fuzzy set and neuron network controllers, that can not be counted 
as reliable enough), we were forced to provide some sort of the linear reference 
model. We chose a generalized describing function method to conduct a piecewise 
linearization of the plant. To do so, an inverse dynamic model was developed, to 
estimate fuel consumption of the engine, given engine speed. Such function was 
developed by the closed-loop optimization versus a direct (fuel to speed) model.

Because of a severe time and workload limits, we restricted our inquiries to 
the step input responses (later we were forced to work only with the raising edge 
steps, because of inappropriate representation of the certain phenomena by the 
model; e.g. engine stall that occurs because of over-cooling when fuel supply 
drops). The inverse model was used to close an optimization loop with a linear 
function under trial in the forward path. The linear function was provided with a 
step input and it’s parameters were tuned to obtain a near zero error at the inverse 
model output. This investigation resulted in the development of the piecewise 
linear model. We found that the good approximation (10% error) can be obtained 
with first-order segments, with all parameters dependent of height, mach number 
and differential of the engine speed. The precision can enhanced by higher order 
linear approximation, but we figured out that third order approximation already 
exceeds all reasonable needs.

The piecewise linear representation of the model allowed us to devise a con-
troller for the engine. We followed a most logical approach - switched bank con-
troller. For each segment on the trajectory we found an inertial stage (integrator + 
first order LPF) and attached them to the model via the analog switch controlled by
special switching algorithm. The evaluation of the real model with this controller allowed us to reduce number of banks (controllers) and to optimize a switching algorithm. The main control criteria were engine speed, surge prevention and overheating control. Engine surge prevention was possible by closing an extra control loop with an internal pressure data, that we assumed to be available from the real engine. If not this loop must be closed with the aid of proper estimator.

We also checked the possibility of the automatic tuning of the controller. It seems possible to obtain a nearly optimal controller, given enough time and computational resources. Pre-optimized, tunable switched bank controller, once designed on the powerful system (up to 200 hours of running on the SGI Origin 2000 for total optimization of tuning constants) will provide a means of efficient real-time control of the plant, requiring minimalistic hardware and few kiloflop/seconds of computational power, yielding the airplane CPU to the more important tasks.