

INTERACTIVE VISUAL ANALYSIS of LARGE SIMULATION ENSEMBLES

Krešimir Matković

VRVis Research Center
Donau-City Str. 1
Vienna, A 1220, AUSTRIA

Denis Gračanin

Department of Computer Science, Virginia Tech
2202 Kraft Drive
Blacksburg, VA 24060, USA

Mario Jelović

AVL AST doo
Strojarska cesta 22
Zagreb, HR 10000, CROATIA

Helwig Hauser

Department of Informatics, University of Bergen
Postboks 7803
Bergen, N 5020, NORWAY

ABSTRACT

Recent advancements in simulation and computing make it possible to compute large simulation ensembles. A simulation ensemble consists of multiple simulation runs of the same model with different values of control parameters. In order to cope with ensemble data, a modern analysis methodology is necessary. In this paper, we present our experience with simulation ensemble exploration and steering by means of interactive visual analysis. We describe our long-term collaboration with fuel injection experts from the automotive industry. We present how interactive visual analysis can be used to gain a deep understanding in the ensemble data, and how it can be used, in a combination with automatic methods, to steer the ensemble creation, even for very complex systems. Very positive feedback from domain experts motivated us, a team of visualization and simulation experts, to present this research to the simulation community.

1 INTRODUCTION

Simulation is an omnipresent support methodology for engineers and scientists across many domains. Many physical phenomena in nature, science and technology can only be explored with the help of simulation. The rapid improvements of modern computation technologies make it possible to simulate systems of increasing complexity. In modern automotive engine design, ever-shorter time to market, together with ever-growing engine complexity, make the use of simulation increasingly important. Strict emission regulations and the shortage of fossil fuels motivate injection system designers in particular. Along with the engine design, accompanying simulation technology also advances to cope with new requirements.

Two main question that a simulation expert usually tries to answer can be summarized as: (1.) How do the dependent variables (outputs) change as the values of the independent variables (control parameters) change? (2.) What values should the independent variables (control parameters) have, so that the dependent variables (outputs) have the desired or expected values. Having a simulation model, it is simple to answer the first question. Answering the second question, however, is far from trivial, as the analytical representation of the simulation model usually cannot be inverted easily. Additionally, the system designers would like to gain insight into the system's behavior. Knowing the optimum setting of the control parameters for a given task, for example, is certainly valuable, but understanding the whole system and the complex interplay of control parameters is something we certainly should strive for in addition.

Recent improvements of computing and storage power, as well as of the simulation technology make a new approach to simulation possible: a phenomenon of interest is not only considered by a single simulation run, but by a multitude of simulation runs of the same simulation model with different settings of control parameters. Such multiple simulation runs are also referred to as ensemble simulation. Ensemble simulation has the great potential to provide a much deeper understanding of the investigated phenomenon, to study the variability and sensitivity of simulation models, and to enable statements about uncertainties that are always associated with such studies.

In this paper, we describe our experience with the ensemble simulation of injection systems (Konyha, Matković, Gračanin, Jelović, and Hauser 2006), (Matković, Gračanin, Jelović, and Hauser 2008), (Matković, Gračanin, Splechtna, Jelović, Stehno, Hauser, and Purgathofer 2014). We describe how interactive visual analysis helps to exploit opportunities that emerge from advanced ensemble simulation. The injection simulation leads to complex outputs far beyond a few scalar values only. Due to a lack of appropriate analysis techniques, the engineers often compute certain scalar features from the time-dependent outputs, and then analyze them as a representation of the complex simulation outputs. This is a proven and valid approach, but as the system complexity grows, and as the analysis questions become increasingly complex, we have to consider original, time-dependent outputs in the analysis, as well. Interactive visualization is a proven technology for analysis of the ensemble simulation data and for studying systems in the light of varying a set of controllable inputs and observing the simulated responses. The methodology described here is not intended to replace conventional, automatic analysis methods that are based on statistics, machine learning, or data mining, for example. Interactive visual analysis represents an efficient additional methodology which helps to gain insights that are otherwise impossible. Only a clever combination of automatic and interactive techniques can solve really complex problems. Besides answering the two high level questions in the analysis, as described above, interactive visual analysis additionally supports the sensitivity analysis, and the knowledge gaining process of the engineer.

2 RELATED WORK

Simulation ensembles have a multi-dimensional parameter space. Analyzing multidimensional parameter spaces has long been of interest to visualization researchers. Sampling of parameter spaces has helped to solve many high-dimensional domain problems, for example, in aircraft design (Shaffer, Knill, and Watson 1998), or in engineering (Stork, Thole, Klimenko, Nikitin, Nikitina, and Astakhov 2008). Some of the authors deal with a large number of simpler responses (1D and 2D), and some focus on 3D responses (Demir, Dick, and Westermann 2014).

There are different strategies for the sampling of the parameter space. Some focus on increasing the number of sampling points (Bachthaler and Weiskopf 2008). Others rely on the fact that in many cases the parameter space for a simulation ensemble can be effectively represented by a set of categorical decisions that can be explored or cross filtered on 1D and 2D projections (Tweedie, Spence, Dawkes, and Su 1996).

Interactive visual analysis (Kerren and Schreiber 2012) — a clever combination of interactive and automatic methods — has been used to analyze simulation results (Doleisch 2007), and time-oriented data in general (Aigner and Tominski 2007). In our previous work (Konyha, Matković, Gračanin, Jelović, and Hauser 2006) we have explored a simulation ensemble by means of interactive visual analysis.

If data analysis is done after a simulation batch is completed, errors invalidating the results of the entire simulation may be detected too late (Parker, Johnson, and Beazley 1997). Computational steering and interactive visualization are visualization paradigms for the computational sciences (Chen, Rine, and Simon 1996) enabling users to interactively steer computations, change simulation parameters and instantly see the simulation results. In this paper we deal with the ensemble steering (Matković, Gračanin, Jelović, and Hauser 2008, Matković, Gračanin, Splechtna, Jelović, Stehno, Hauser, and Purgathofer 2014). We do not steer a single simulation, but the creation of an entire ensemble. Such an approach is possible when a single simulation run can be computed fast.

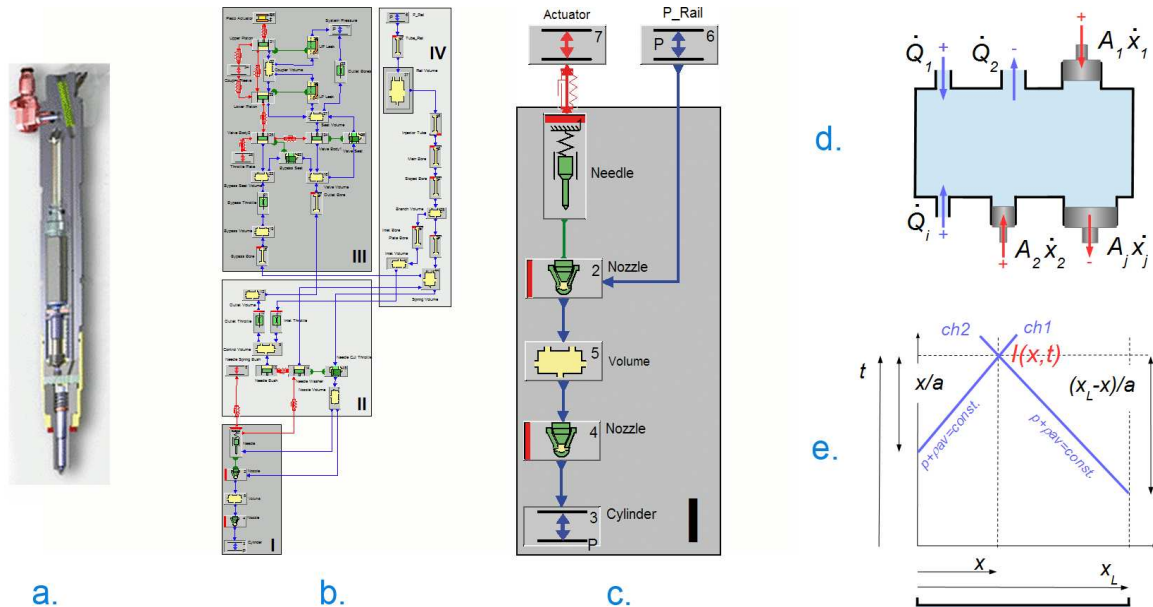


Figure 1: **a.** A photograph of a real fuel injector. **b.** AVL Boost-Hydsim simulation model of the injector. **c.** One of the simplified models of the same injector used during the analysis. **d.** A diagram of the capacitive component. **e.** Two characteristic lines of the pipe component.

3 COMMON RAIL INJECTION SIMULATION

The common rail injection system represents the standard injection system for Diesel car engines (Boecking, Dohle, Hammer, and Kampmann 2005), (Boehner and Hummel 1997). It operates at a very high pressure level, with an electronic control unit which controls the fuel delivery, injection timing, injection pressure, and rate of injection, for multiple injection strategies. The modern common rail makes it possible for Diesel cars to achieve a level of performance and driving comfort similar to those of gasoline powered models, with reduced fuel consumption and low exhaust emissions.

The main principle of a common rail system is the use of a high-pressure rail, common to all cylinders. The high pressure in the rail is used to precisely inject the fuel into the cylinders. Electronically controlled actuators open and close the injectors. Sometimes, in one cycle, the main injection is preceded by a pilot injection, or even several of them. All this happens at least several hundred times per second.

Due to high pressures and quick changes in the system, a modern common rail injection system operates in a condition which cannot be described sufficiently precise using classical fluid mechanics. The Diesel fuel cannot be considered incompressible when exposed to high pressure changes in short time periods. At high pressure, the fluid density, the module of elasticity, and the speed of sound are significantly altered. Furthermore, in a common rail system each cylinder and injector is influenced by the others through the rail. This requires a careful rethinking of traditional system design.

We explored a single injector design, as well as the complete injection system for four cylinders. Due to the system complexity, a simulation ensemble is computed, and results are analyzed then. The main goal is to tune a common rail injection system for various modes of operation.

3.1 Simulation Model

The common rail system is a complex system consisting of several subsystems. These include electric, magnetic, mechanical, and hydraulic subsystems. From the fluid flow perspective, the common rail system may be described by the following types of components:

- Resistive components
- Capacitive components
- Pipe-type components (dynamic of pressure wave in hydraulic lines is taken into account)
- Hydro-mechanical components (the pressure is transformed into force)

The simulation software, AVL Boost-Hydsim in our case (AVL 2015), is used to simulate the common rail injector. Figure 1a shows a photograph of the real fuel injector, and Figure 1b shows the simulation model for a single injector. The engineer starts with the modeling of the injector. Using a limited set of available components, the simulation model is composed. The components are connected with each other. The model creation is a crucial step in the simulation, and engineer's experience and problem understanding play an essential role in this design phase. In this paper we assume that the model is correct, and we deal with model parameters tuning and optimization. Once the model is created, control parameters for each component have to be set. There are components of different complexity and the number of control parameters varies across different components. Two types of components — the capacitive component and the pipe component — contribute most to the phenomena under consideration. Therefore, we describe them in more detail in the following.

3.1.1 Capacitive Component

The capacitive or volume element is not restricted to any geometric shape. A schematic of the element is shown in Figure 1d. The pressure in the volume at isothermal flow is calculated from the continuity equation:

$$\dot{p}(t) = \frac{E}{V(x)} \left(\sum_{i=1}^{n_1} \dot{Q}_i + \sum_{j=1}^{n_2} A_j \dot{x}_j \right) \quad (1)$$

where $\dot{p}(t)$ is pressure time derivative, E is bulk modulus of the fluid and $V(x)$ is the actual volume. The actual volume is calculated from the following equation:

$$V(x) = V_0 - \sum_{j=1}^{n_2} A_j \dot{x}_j \quad (2)$$

3.1.2 Pipe Component

The pipe component represents a pipe of a circular cross-section. The AVL Boost-Hydsim (AVL 2015) user may choose between different, advanced, pipe models such as, d'Alembert, Laplace, Characteristic, or Godunov and Mac-Cormack, for example. In this paper our interest is not to investigate the shock-wave propagation along any pipe, but only its effect on the pipe ends. Therefore we use a pipe model of a medium complexity, so called Laplace line.

Navier-Stokes equations (the law of conservation of momentum) and continuity equation (the mass conservation law) are simplified and given in the following form:

$$\rho \frac{\partial v}{\partial t} + \frac{\partial p}{\partial x} = \mu \left(\frac{\partial^2 v}{\partial r^2} + \frac{1}{r} \frac{\partial v}{\partial r} \right) \quad (3)$$

$$\frac{\partial \rho}{\partial t} + \rho \frac{\partial v}{\partial x} = 0 \quad (4)$$

The system has three unknowns: the x coordinate of the infinitesimally small control volume, the pressure, and the density. Three independent equations are required to solve them. The third equation is the equation of state. It may be derived from the linear acoustic theory:

$$\frac{\partial p}{\partial \rho} = a^2 = const. \quad (5)$$

where a is the velocity of sound in the fluid. After transformations, equations are reduced to the equations of Allievi (Seifert 1962):

$$\frac{\partial p}{\partial t} + \rho a^2 \frac{\partial v}{\partial x} = 0 \quad (6)$$

$$\frac{1}{\rho} \frac{\partial p}{\partial x} + \frac{\partial v}{\partial t} = R \quad (7)$$

where R is the friction function. The d'Alembert method yields the solution of the Allievi equations without friction. State variables at the point $I(x,t)$ can be calculated from the coordinates of line ends (the lines are shown in Figure 1e.):

$$ch1 : p(x,t) + \rho av(x,t) = p\left(0, t - \frac{x}{a}\right) + \rho av\left(0, t - \frac{x}{a}\right) \quad (8)$$

$$ch2 : p(x,t) - \rho av(x,t) = p\left(x_L, t - \frac{x_L - x}{a}\right) - \rho av\left(x_L, t - \frac{x_L - x}{a}\right) \quad (9)$$

4 ENSEMBLE SIMULATION ANALYSIS

The main idea of the ensemble simulation is to compute the simulation results for different combinations of control parameters using the same simulation model. Once the results are computed, the user explores them using a coordinated multiple views system. The coordinated multiple views include at least two views depicting different control parameters and output values. The most often used views include the scatterplot, the histogram, parallel coordinates, or a curve view. The views support linking and brushing – the user interactively selects some items (simulation runs in our case) in one view, and the corresponding items all other views are highlighted. The user can now move the brush and observe what happens with the corresponding items. As the number of control parameters increases, a dense sampling of the parameter space results in a prohibitively large ensemble. Interactive visual steering solves the problem. Instead of densely covering the multi-dimensional parameter space, the space is sampled coarsely, and, as potentially interesting parts are identified by means of the interactive exploration, additional runs are initiated from the visualization. If the number of the control parameters further increases, interactive visual steering is not sufficient any more. The identification of interesting areas in the parameter space becomes too tedious. A combination with automatic methods can help with this problem.

We describe all three cases in the example of common rail injection. We start with the tuning of a single injector. The number of parameters we vary is low, and we can generate the whole ensemble in advance. In the next step we then vary eleven parameters. If there are ten variations of each (which is realistic), and we opt for a full factorial sampling of the parameter space, the resulting ensemble would have 10^{11} simulation runs. Although our simulation is fast, i.e. we compute 10 simulations per minute, it is practically impossible to compute such a large ensemble. We use interactive visual steering in order to cope with the system complexity. Finally, when we tune the whole four cylinder injection system, we need the support from automatic methods, as well.

4.1 Interactive Visual Analysis of a Single Injector

There are three main factors that influence the injection behavior: the nozzle geometry, the injection pressure, and the timing of the valve opening and closing procedures. The nozzle geometry cannot be changed at run-time, and it is engineered by injector manufacturers. Therefore, we focus on the remaining two factors in the analysis.

Accordingly, the independent variables in our ensemble are related to injection pressure and the injector valve timings. The injection pressure is controlled by the injection pressure modulation device. The pressure on the injectors inlet is described by three independent variables: P_{low} , P_{high} , and dT_p (Figure 2a., left).

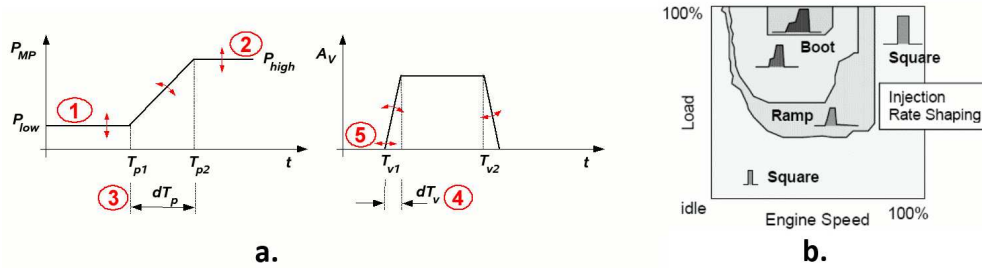


Figure 2: **a.** Three scalar control parameters, describing the pressure at the injectors inlet, and two scalar features, describing the actuator time signal. **b.** Characteristic shapes of the main injection curves for different combinations of engine speed and load.

Table 1: Five control parameters varied.

Control Parameter	Symbol	Number of variations
Low pressure on the injector inlet	P_{low}	5
High pressure on the injector inlet	P_{high}	5
Time interval of modulated pressure increase on the injectors inlet	dT_p	5
Time interval of the injector valve opening and closing	dT_v	5
Injector valve opening time	T_{v1}	7

The injection timing is controlled by the injector valve actuator. We model the timing by opening and closing times and velocities, which result in two additional independent variables: dT_v and T_{v1} (Figure 2a., right). There are five independent variables – control parameters, in total. Table 1 shows the varied control parameters and the number of variations for each of them. The total number of variations, i.e. the number of simulation runs we have computed, equals $5^4 \times 7$, or 4375 runs. For each simulation run we compute following, time-dependent and scalar response variables: injection rate $Q_{inj}(t)$, injection pressure $P_{inj}(t)$, needle lift $A_n(t)$, amount of fuel injected during pilot injection Q_p , amount of fuel injected during main injection Q_m , amount of fuel flowing back to the fuel tank Q_{vo} , needle opening velocity V_{open} , needle closing velocity V_{close} , spray penetration depth L_p , and average injection power P_{ia} .

Once the ensemble is computed, the analysis can start. We first explore how the changes in control parameters influence the output values. We depict two of the control parameters using a scatterplot, and the response variables under consideration using a curve view and a scatterplot, as shown in Figure 3a. We brush the upper left corner in the scatterplot (high values of dT_p and low values of T_{v1}), and move the brush towards lower left corner. We observe changes in other views as we move the brush. Figure 3b. shows the brushes' positions (all brushes are superimposed in this figure to save space), and Figures 3c to 3g show what happens with one time dependent response variable $Q_{inj}(t)$. We examine several responses in the same way, at once. By doing so, we can comprehend the system behavior very fast.

The next step is the model reconstruction. We wish to identify control parameters that result in a desired model behavior. The optimum injection curve shape depends on the engine operating point. Depending on engine speed and load, the desired shape varies. Figure 2b. shows desired curve shapes for different operating points. We depict injection curves in a curve view and use several line brushes to "specify" the desired curve shape. The line brush selects all curves that cross the drawn line. By combining several lines using Boolean operators we can specify a shape flexibly and quickly. Figure 4 shows an example. We used five line brushes in order to specify a desired boot shape. The control parameters are shown in two scatterplots and one histogram. We can see that, in order to achieve the boot shape, we have to set T_{v1} to the lowest value, dT_p and P_{low} to the upper half of the available range, P_{high} to the lower half, and, as it can be seen in the histogram, dT_v can have any value.

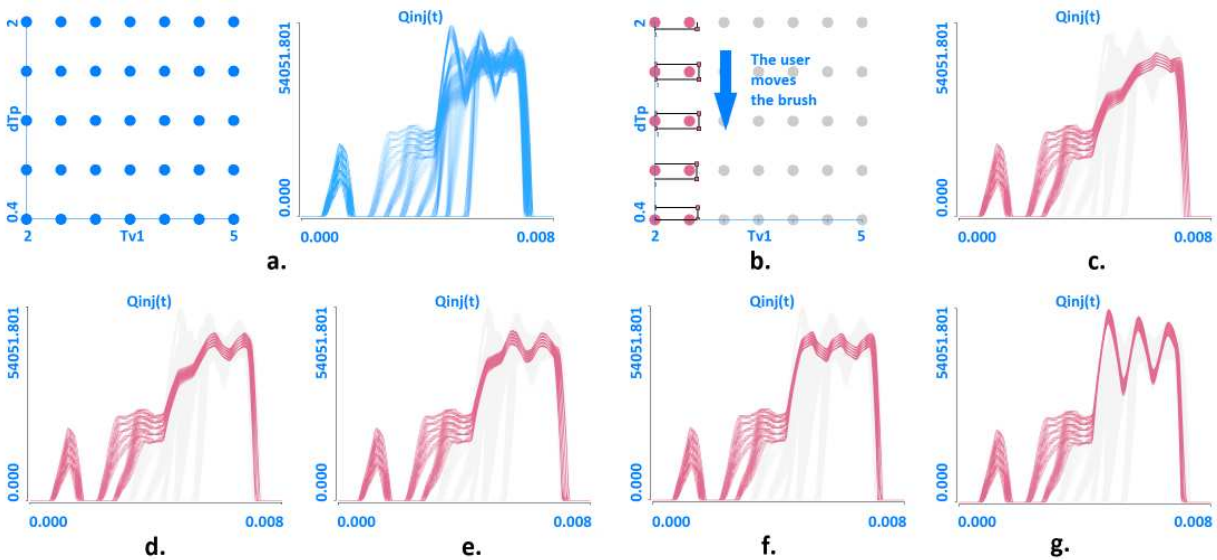


Figure 3: Exploring changes in responses as the control parameters change. **a.** The minimum configuration for IVA. The scatterplot shows two control parameters, and the curve view shows one of the time dependent response variables. **b.** The user brushes the upper left corner in the scatterplot and moves the brush downwards. **c. to g.** As the brush moves the corresponding curves are highlighted. The user knows that dT_p is decreasing, while T_{v1} remains low, and can focus on the responses only.

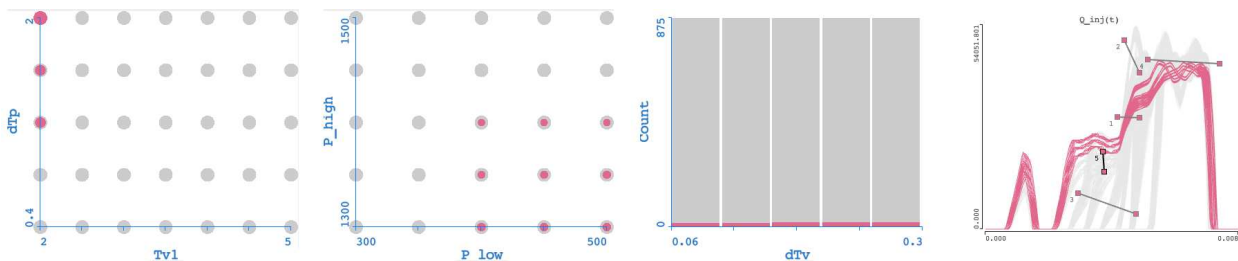


Figure 4: Model reconstruction – the user identifies control parameters for a desired injection shape. A desired shape is selected by means of several line brushes in the curve view. The scatterplots and the histogram show the corresponding control parameters.

Finally, sensitivity analysis has to be performed. In the case of interactive visual analysis, simple brushing of the parameters in parallel coordinates, or in a scatterplot, e.g., immediately shows the corresponding outputs. In this way, the user gains insight into parameters sensitivity as well. Of course, all findings should be formally confirmed once the interactive analysis is finished.

Due to the analysis, we gained valuable insight into the simulation ensemble and into the fuel injection process itself. The most important findings can be summarized as:

- The amount of injected fuel in the main injection stage can be controlled by adjusting P_{high} .
- The amount of pilot injection is controlled mostly by P_{low} , but dT_v also influences it.
- A right choice of dT_p and T_{v1} is the key to achieving the desired injection shapes for various engine operating conditions.
- When pressure increases too fast on the injectors inlet, the resulting wave can be reflected into the fuel line, which impairs our control over the injections shape.

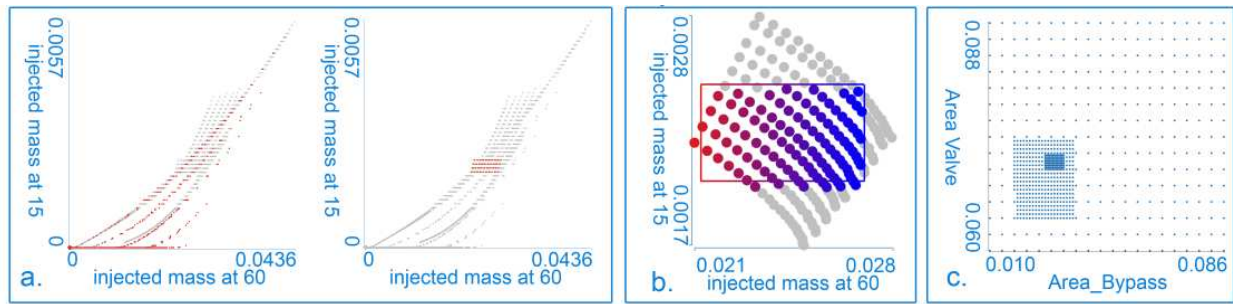


Figure 5: **a.** During visual steering values of response parameters change as the analysis proceeds. **b.** A zoomed-in area of the same features at the end of the analysis. **c.** A scatterplot showing two control parameters. Two refinements during the steering process are clearly visible.

4.2 Interactive Visual Steering

So far we used the visualization and simulation separately. This is sufficient if the system under consideration has a limited set of control parameters. If we want to vary more parameters, a more advanced approach is needed. In the next stage of the analysis, we then tune eleven different parameters.

The interactive visual steering makes it possible to define new simulations from the visualization tool. The visualization tool is used for the exploration and analysis, and, at the same time, to initiate new simulation runs. That enables the user to generate new simulation runs and to refine (or to filter) the simulation ensemble. We provide two basic operations at a high level: refining an existing ensemble and initiating a new ensemble. The ensemble refinement does not change the ensemble structure, we keep the control parameters as they are, we just compute new simulation runs, for newly defined combinations of control parameters. In a case of new ensemble initiations, the available control parameters change, we refine or coarsen the model itself. The whole process is an interactive, iterative loop, where each step is executed multiple times. At an individual step, the exploratory analysis process, as described in the previous subsection, is pursued.

During the injector tuning process we used four models. The simplest one consists of the first block only (Figure 1c), and the rest is represented by an ideal actuator. Additional modules are consecutively added, as new ensembles are initiated. The second level includes blocks one and two, the third level adds the block three, and so on. The blocks are depicted in Figure 1 b. The idea is to tune the coarse model first, and to limit the possible range of the control parameters based on this analysis step. Of course, if further steps show that the control parameters yield undesired results, the values are altered. In most cases, the control parameter ranges estimated using a coarse model will be a good indicator for final parameter ranges. During the analysis, at each complexity level, additional runs are computed. Figure 5c shows a scatterplot depicting two control parameters. Two refinements are clearly visible. These refinements were initiated from the visualization itself. Figures 5a and 5b show how the values of two response variables, injected mass at crankshaft angle of 15, and injected mass at the crankshaft angle of 60, vary during the analysis. At the final stage, the range is narrowed down (Figure 5b) and the solution is selected among this points.

The interactive visual ensemble steering makes it possible to analyze more complex models. It should be observed as an outer loop around the interactive visual analysis as described in the previous section. For a larger number of control parameters, computing the whole ensemble in advance, using a reasonable sampling density, is usually not possible.

4.3 Hybrid Visual Steering

After individual injectors are tuned, the whole injection system has to be optimized. Figure 6 shows the simulation model of the whole system, and a photograph of a real injection system. The common rail, the

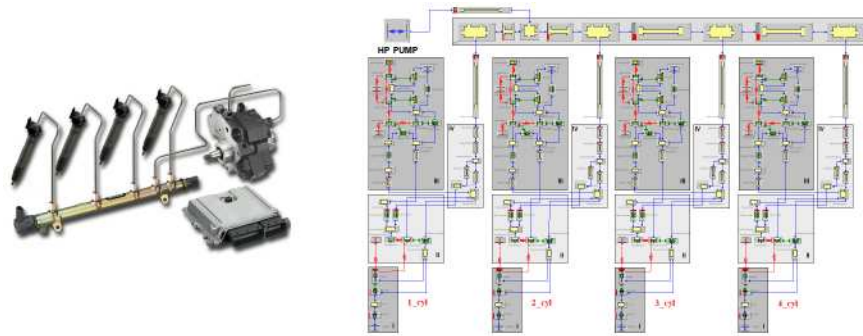


Figure 6: The complete injection system and the corresponding simulation model.

high pressure pump, pipelines and four injectors are shown in the photo. As the number of parameters increases further, interactive visual steering alone is not sufficient. In order to cope with such complex systems, we propose hybrid visual steering (Matković, Gračanin, Splechna, Jelović, Stehno, Hauser, and Purgathofer 2014). As the parameter space is high-dimensional, identifying the areas of interest for a refinement becomes tedious. An automatic optimization, based on regression models, can be used to support the user in the refinement process. The main idea is to compute a regression model, representing the whole system, and to use it to automatically compute different optimum values. The user can request (from the visualization, again) the point of the minimum consumption, for example. The system computes the point using a regression model. The regression model is only an approximation. Two main reasons contribute to the inaccuracy of the automatically computed optimum: **(1)** The optimum is based on the regression model which is an approximation. **(2)** The regression model is computed using scalar responses only. As we have time-dependent responses, they have to be represented with several scalar features prior to the regression model building. The regression model is built using a limited set of scalar features, which represent an approximation of time-dependent results. This introduces additional inaccuracy. The visualization provides a solution, again. Once optimum based on the regression model is computed, the results (scalar and time-dependent) are computed using the simulation. Additionally, we run the simulation for a set of points in the neighborhood of the proposed solution. The user sees the results, and can steer the process in the desired direction. Additional runs can be computed, and the whole process is executed in a loop. Several different optimum values (for different goals) are computed during the analysis process. The engineer has to meet many compromises and decide on the final solution, eventually. In order to realize such a hybrid steering framework, several basic components have to be integrated into a unified system. These components include:

- **Design of Experiment (DOE) Component:** supports the specification of sampling points (designs) in the multi-dimensional parameter space.
- **Simulation Component:** simulates the phenomena under consideration.
- **Analysis and Exploration Component:** the key component supporting advanced user interaction, interactive feature extraction, the visualization of complex and scalar simulation results, and the interactive selection of new design points. It also controls regression model building, evaluates it, and shows optimization results.
- **Regression Model Building Component:** builds a regression model from the simulation ensemble using scalar control parameters, scalar responses, and scalar features of the time dependent responses.
- **Automatic Optimization Component:** computes an optimum using the regression model, subject to the interactively specified constraints.

Figure 7 illustrates the whole process and the basic building blocks. The Figure also indicates parts that belong to the previously described analysis approaches: interactive visual analysis and interactive visual

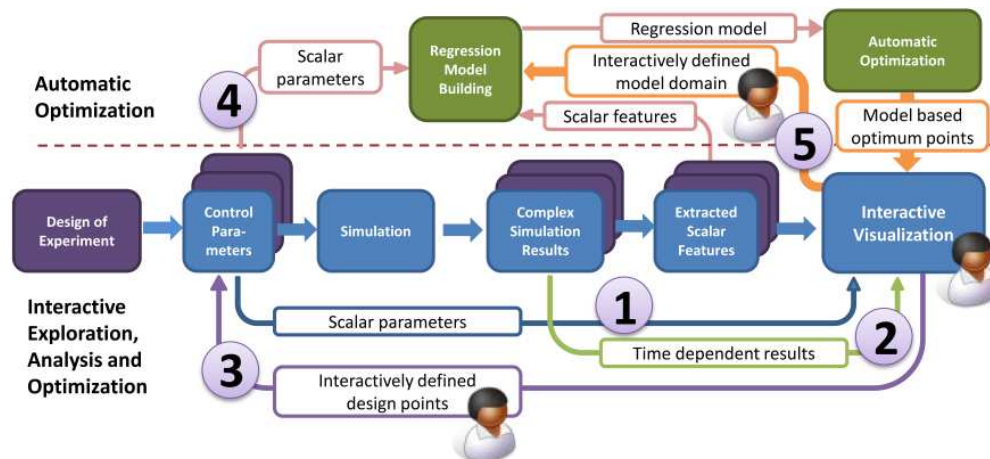


Figure 7: A diagram of the hybrid steering process. The numbers one to five indicate five basic loops ranging from simple scalar values exploration (one) to hybrid steering (five).

steering. Five loops, marked with numbers one to five can be seen. The design of experiment component is always needed. It defines the initial ensemble, and the simulation tool computes responses for a set of combinations of control parameters. The first loop deals with scalar responses only. It includes interactive visual analysis and uses control parameters and scalar responses to analyze a precomputed ensemble. The second loop adds a support for time dependent responses. In the third loop, new simulation runs can be initiated from the visualization. The first and the second loop are still active. The fourth loop represents an alternative approach. We use scalar parameters and scalar response to build a regression model. The model can be used for automatic optimization. The fifth loop, integrates all loops, so that we can use results from the automatic optimization as guidelines during interactive visual steering approach. It represents the hybrid visual steering approach.

In order to tune the whole injection system we use hybrid steering. We vary the pipeline's geometries and the common rail volume in order to achieve an optimum operation of the fuel injection system for all engine speeds (at the level of a separated branch per cylinder, and in relations between individual branches). We vary the following parameters:

- The length and diameter of the high-pressure pipes. All high-pressure pipes during the first phase of investigation are identical to each other.
- Volume (size) of the common rail and initial pressure.
- The volume at the entrance to the injector itself.
- Start time for the first pilot injection. The time lag between pilot and main injection is kept constant.

We research the influence of individual variations and combinations at two levels. At the level of each cylinder we are interested in:

- Differences among injection pressures of the individual pilot injections.
- Differences among the amounts of the injected fuel during the individual pilot injections.
- Needle opening and closing velocities for two pilot injections and the main injection.
- Damping of pressure oscillations that can occur within the high-pressure pipe

And at the whole system level, when all injectors are taken into consideration, we focus on:

- Differences among injection pressures between individual cylinders.
- Injected amounts of fuel between individual cylinders.

Hybrid visual steering made it possible to analyze the whole system successfully. During the analysis engineers commented on advantages as compared to traditional approaches. One engineer commented: "The suggestions of where to refine the parameter space based on optimization speeds up the steering. The shown model accuracy indicates the quality of the optimization. Seeing all runs all the time is simply unmatched in a conventional workflow. I would estimate a speedup of at least an order of magnitude when designing complex systems." He was also pleased with the integrated system: "I could never set up optimization so fast. I also see all results together with the initial runs." Due to the limited space we omit a longer evaluation here. Interested readers are referred to our previous work.

5 CONCLUSION

As the simulation develops, it is necessary that analysis methodology follows the development. Simulation ensembles, a recent trend in the simulation offers new possibilities for the exploration and analysis. We describe our long-term experience in the analysis and steering of the simulation ensembles by means of interactive visual analysis. This is not intended to replace conventional approach, it makes it possible to gain new insights in the system behavior. For very complex systems, with a multi-dimensional parameter space, only a clever combination of visualization and automatic methods (such as in our hybrid visual steering, e.g.) can lead to better understanding of the phenomenon under consideration. Very positive feedback from domain experts documents the usefulness of the presented analysis approach. In spite of all our efforts it was not possible to precisely quantify speed-up compared with conventional analysis approach. A more formal evaluation, as well as an evaluation in other domains is subject to future work.

ACKNOWLEDGMENTS

We thank Rainer Splechtna from VRVis and Zoltan Konyha from AVL for numerous discussions. Part of this work was done in the scope of the K1 program at the VRVis Research Center.

REFERENCES

- Aigner, W., and C. Tominski. 2007, 9–12. "Towards a Conceptual Framework for Visual Analytics of Time and Time-Oriented Data". In *Proceedings of the 2007 Winter Simulation Conference*, 721–729: IEEE.
- AVL 2015. "AVL List GmbH". <http://www.avl.com/>. [last accessed 13 July 2015].
- Bachthaler, S., and D. Weiskopf. 2008. "Continuous Scatterplots". *IEEE Transactions on Vis. and Comp. Graphics* 14 (6): 1428–1435.
- Boecking, F., U. Dohle, J. Hammer, and S. Kampmann. 2005. "Passenger Car Common Rail Systems for Future Emissions Standards". *MTZ worldwide* 66 (7–8): 552–557.
- Boehner, W., and K. Hummel. 1997. "Common Rail Injection System for Commercial Diesel Vehicles". *SAE Transactions* (SAE 970345).
- Chen, J. X., D. Rine, and H. D. Simon. 1996, Winter. "Advancing Interactive Visualization and Computational Steering". *IEEE Computational Science & Engineering* 3 (4): 13–17.
- Demir, I., C. Dick, and R. Westermann. 2014, December. "Multi-Charts for Comparative 3D Ensemble Visualization". *IEEE Transactions on Vis. and Comp. Graphics* 20 (12).
- Doleisch, H. 2007, 9–12. "SimVis: Interactive Visual Analysis of Large and Time-Dependent 3D Simulation Data". In *Proceedings of the 2007 Winter Simulation Conference*, 712–720: IEEE.
- Kerren, A., and F. Schreiber. 2012, 9–12. "Toward the Role of Interaction in Visual Analytics". In *Proceedings of the 2012 Winter Simulation Conference*, 1–13: IEEE.
- Konyha, Z., K. Matković, D. Gračanin, M. Jelović, and H. Hauser. 2006, November/December. "Interactive Visual Analysis of Families of Function Graphs". *IEEE Transactions on Vis. and Comp. Graphics* 12 (6): 1373–1385.

- Matković, K., D. Gračanin, M. Jelović, and H. Hauser. 2008, November-December. “Interactive Visual Steering – Rapid Visual Prototyping of a Common Rail Injection System”. *IEEE Transactions on Vis. and Comp. Graphics* 14 (6): 1699–1706.
- Matković, K., D. Gračanin, R. Splechtina, M. Jelović, B. Stehno, H. Hauser, and W. Purgathofer. 2014. “Visual Analytics for Complex Engineering Systems: Hybrid Visual Steering of Simulation Ensembles”. *IEEE Transactions on Visualization and Computer Graphics* 20 (12): 1803–1812.
- Parker, S. G., C. J. Johnson, and D. Beazley. 1997, October-December. “Computational Steering: Software Systems and Strategies”. *IEEE Computational Science & Engineering* 4 (4): 50–59.
- Seifert, H. 1962. *Instationäre Strömungsvorgänge in Rohrleitungen an Verbrennungskraftmaschinen: Die Berechnung nach der Charakteristikenmethode*. Springer Berlin Heidelberg.
- Shaffer, C. A., D. L. Knill, and L. T. Watson. 1998. “Visualization for Multiparameter Aircraft Designs”. In *Proceedings of the Conference on Visualization '98*, 491–494. IEEE Computer Society Press.
- Stork, A., C.-A. Thole, S. Klimenko, I. Nikitin, L. Nikitina, and Y. Astakhov. 2008. “Towards Interactive Simulation in Automotive Design”. *The Visual Computer* 24 (11): 947–953.
- Tweedie, L., R. Spence, H. Dawkes, and H. Su. 1996. “Externalising Abstract Mathematical Models”. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 406–ff. ACM.

AUTHOR BIOGRAPHIES

KREŠIMIR MATKOVIĆ is a senior researcher at VRVis Research Center in Vienna. He is interested in extending visual analysis technology to challenging heterogeneous data, in particular to a combination of multi-variate data and more complex data types, such as functions, e.g. He also focuses his research on developing a structured model for visual analysis which supports a synergetic combination of user interaction and computational analysis. He teaches at TU Vienna and at University of Zagreb. He is a member of ACM, Eurographics, and IEEE Computer Society. His email address is Matkovic@VRVis.at.

DENIS GRAČANIN received the BS and MS degrees in electrical engineering from the University of Zagreb, Croatia, in 1985 and 1988, respectively, and the MS and PhD degrees in computer science from the University of Louisiana at Lafayette in 1992 and 1994, respectively. He is an Associate Professor in the Department of Computer Science at Virginia Tech. His research interests include virtual reality and distributed simulation. He is a senior member of ACM and IEEE and a member of AAAI, APS, ASEE and SIAM. His email address is gracanin@vt.edu.

MARIO JELOVIĆ is a skill team leader in the Powertrain Simulation Tool Development group within the AVL Advanced Simulation Technologies department. He has been actively participating in research and development within the AVL company for more than 15 years. During his professional career he has been involved in many projects in academia and industry in the area of 1D computational fluid dynamics, as a project manager or a team member. He coauthored several scientific, as well as technical papers. One of these papers was judged by SAE International to be among the most outstanding SAE technical papers of 2010. His email address is mario.jelovic@avl.com.

HELWIG HAUSER is professor at the University of Bergen, Norway, where he is leading a research group on visualization since 2007. Before moving to Norway and since 2003, Helwig Hauser was the scientific director of the VRVis Research Center in Vienna, Austria. Earlier, he was assistant professor at the Vienna University of Technology, from which he also received his graduate and doctoral degrees (in 1994 and 1998) as well as his habilitation (2003). Helwig Hauser received several awards, including the biannual Heinz-Zemanek Award in computer science from OCG in 2006 and the Dirk Bartz Prize for visual computing in medicine in 2013. Helwig Hauser is an active member of the international visualization community. His email address is Helwig.Hauser@UiB.no.