Use of Neural Networks for Approximation of Response Functions

Igor Grešovnik

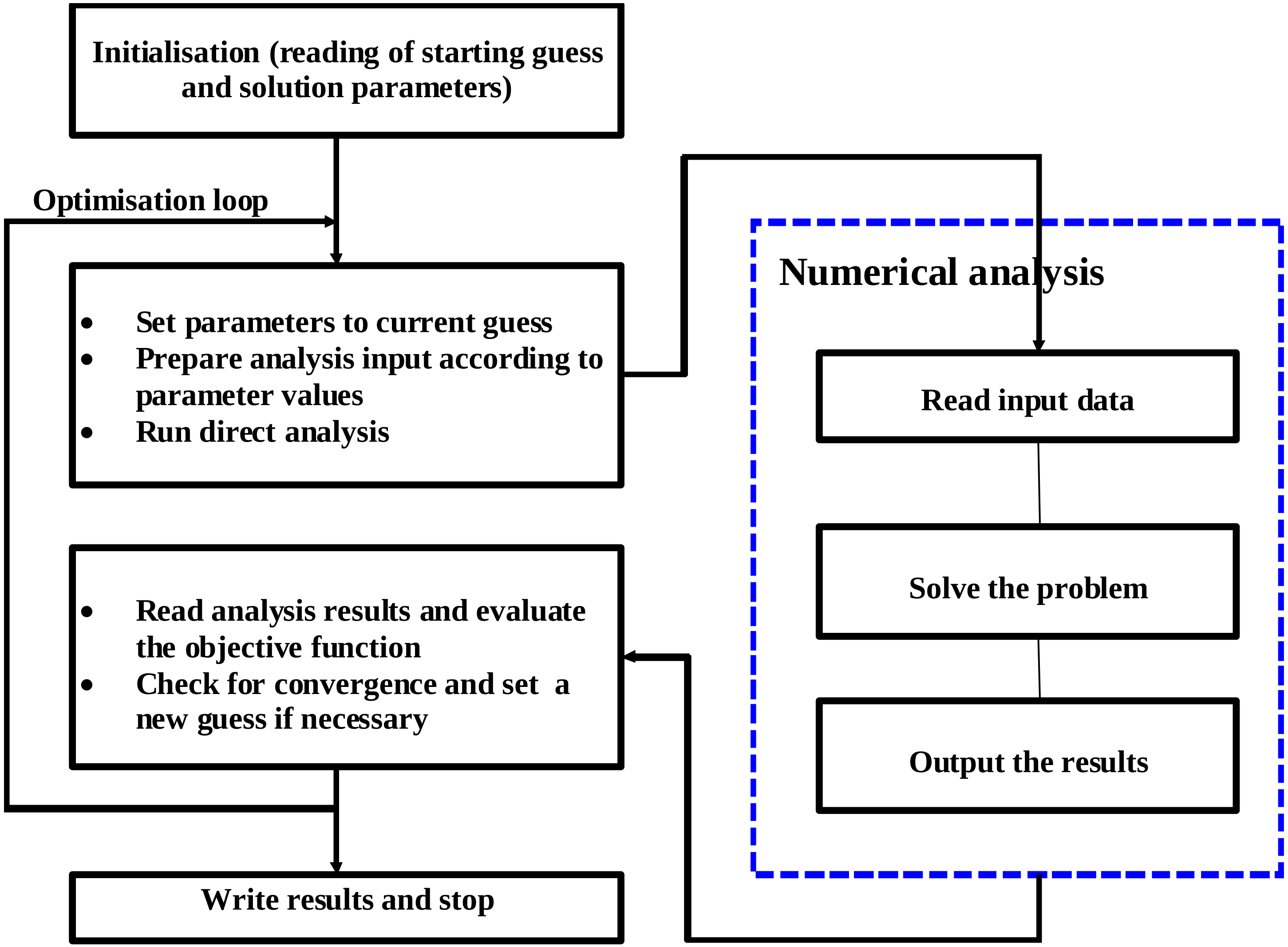
June 2011

Thes presentation shows use of neural networks in solution of optimization problems.

Optimization Problems – Formulation

minimise	$f(\mathbf{x}), \quad \mathbf{x} \in \mathbb{R}^n$
subject to	$c_i(\mathbf{x}) \le 0, \ i \in I$
and	$c_j(\mathbf{x}) = 0, \ j \in E$,
where	$l_k \le x_k \le u_k, \ k = 1, 2, \ldots, n$.

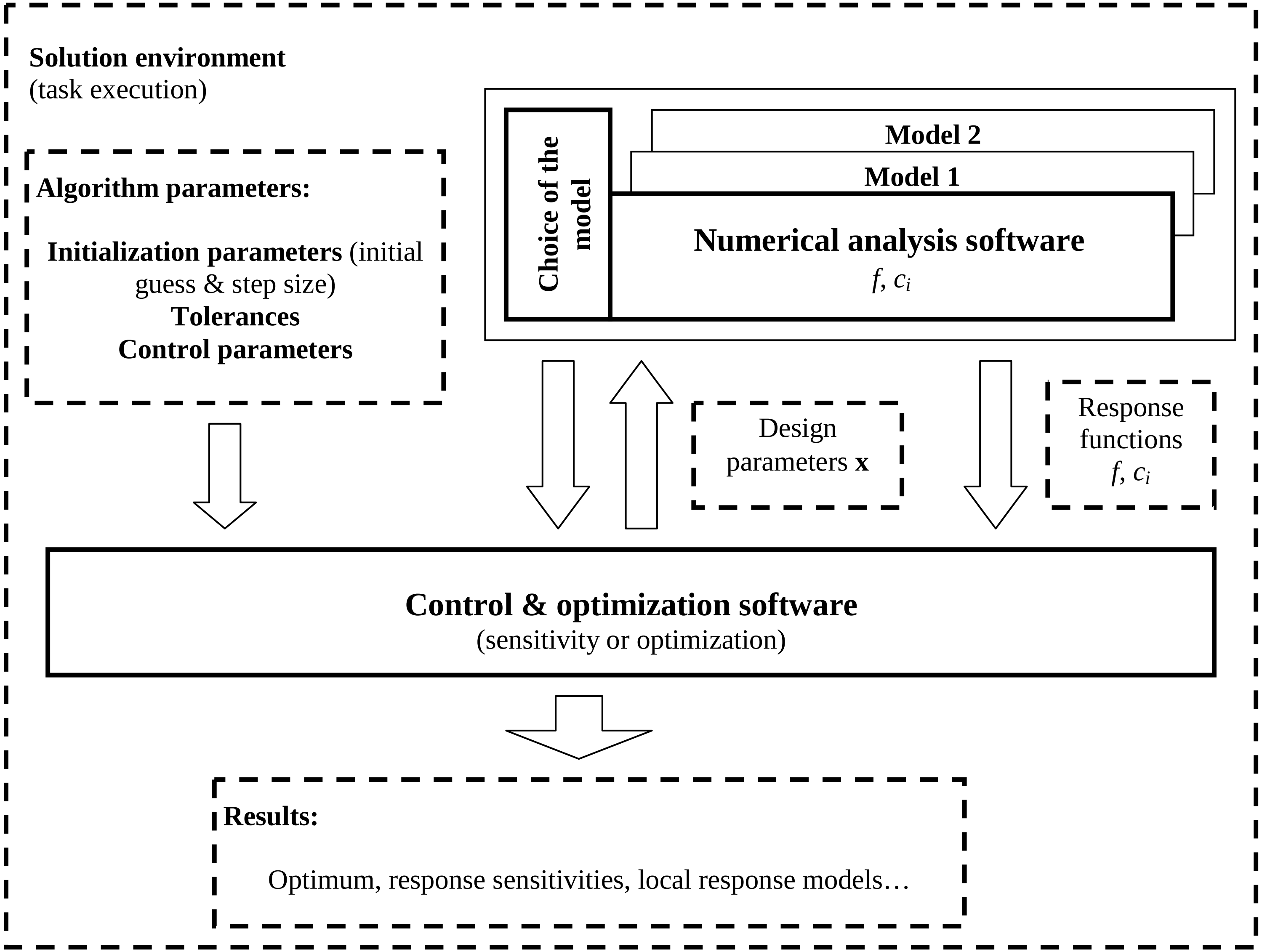
Optimization Problems – Solution Scheme



Optimization Problems – Solution Scheme

1. Take current optimization parameters
2. Prepare numerical model according to parameters
3. Run numerical simulation of the process
4. Extract the relevant quantities from simulation results
5. From measured data
    - Read result file
    - Extract relevant data
6. Calculate the response functions and eventually their gradients (in our case the discrepancy function *f*)
7. Store the response functions in output arguments and return

Integrated Optimization Platform



File Format for Data Exchange

Analysis input file:

```
{ { p1, p2, … }, { reqcalcobj, reqcalcconstr, reqcalcgradobj,
      reqcalcgradconstr }, cd }
```

Analysis output file:

```
{
  { p1, p2 ... },
  {
    calcobj, obj,
    calcconstr, { constr1, constr2, ... },
    calcgradobj, { dobjdp1, dobjdp2, ... },
    calcgradconstr,
    {
      { dconstr1dp1, dconstr1dp2, ... },
      { dconstr2dp1, dconstr2dp2, ... },
      ...
    },
    errorcode
  },
  { reqcalcobj, reqcalcconstr, reqcalcgradobj, reqcalcgradconstr }
  < , { ind1, ind2, ... }, { coef1, coef2, ... }, defdata >
}
```

Neural Networks: Response Approximation

- **Provides approximate relation between process parameters and outcomes**

Π:

π_1
π_2
π_3
π_4
…
…
…
…
…
$\pi_{N\pi}$

⇒

Trained neural network

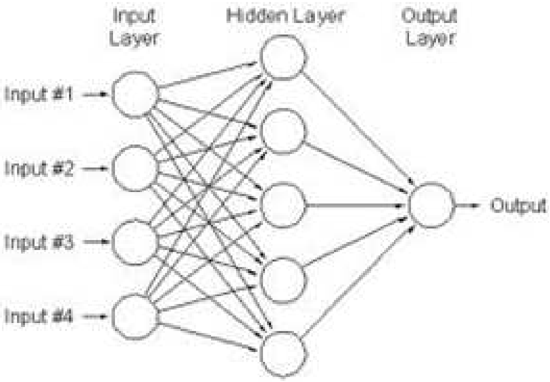
(known layout and known weights w_i)

⇒

Ω:

ω_1
ω_2
…
…
…
$\omega_{N\omega}$

Neural Networks: Training



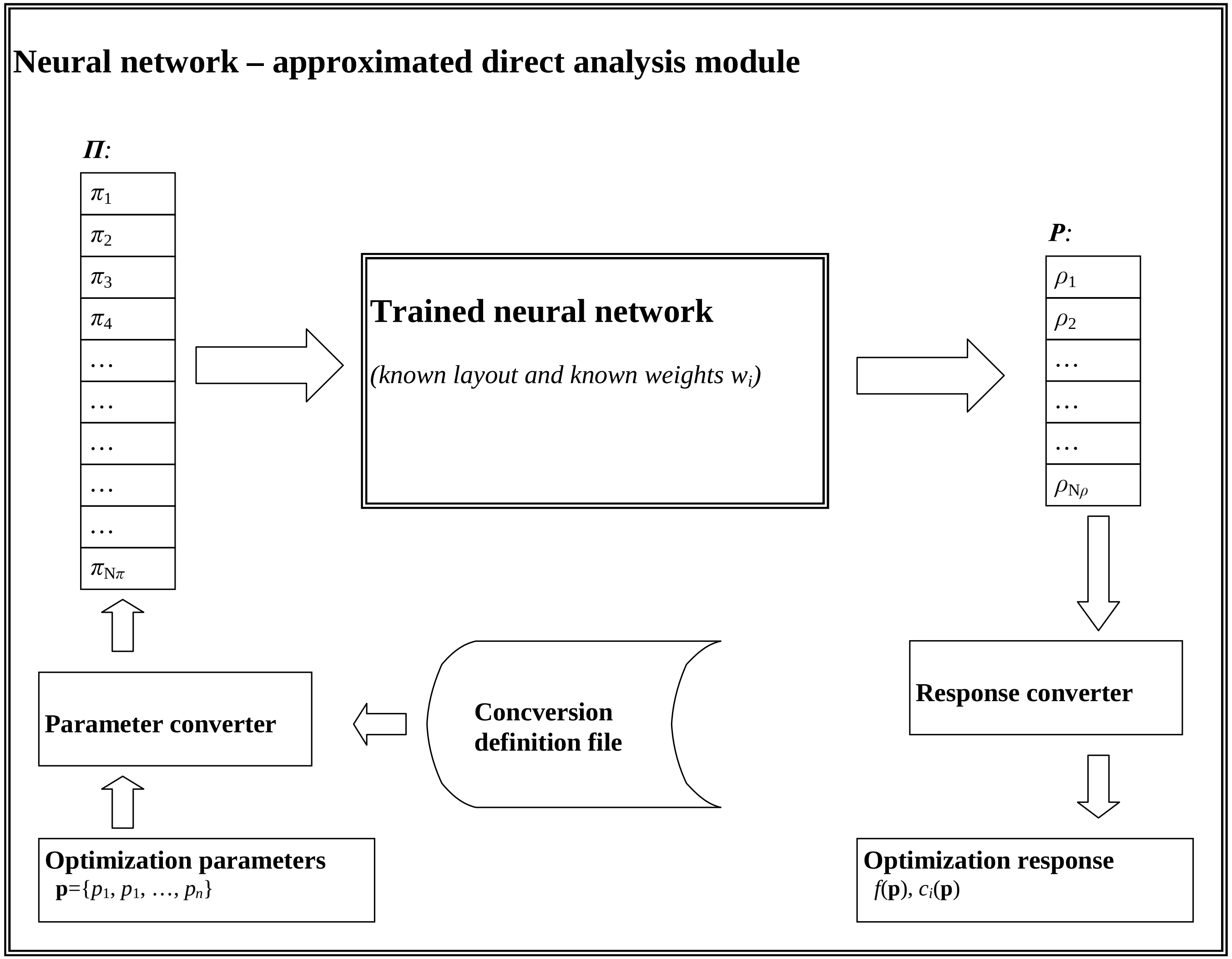
Training data

(Π_1, Ω_1)
(Π_2, Ω_2)
(Π_3, Ω_3)
(Π_4, Ω_4)
…
…
…
…
…
(Π_5, Ω_5)

Trained neural network

(known layout and known weights w_i)

Neural Networks: Direct Analysis Surrogate



Neural Networks: Direct Analysis Surrogate

Conversion definition file

***Π**:*	***active flag**:*	***Corresponding opt. parameter**:*	***Default value:**:*
π_1	*yes*	*1*	*3.56*
π_2	*no*	*0*	*1.2e7*
π_3	*no*	*0*	*109.3*
π_4	*yes*	*2*	*24.5*
…	…	…	…
…	…	…	…
…	…	…	…
…	…	…	…
…	…	…	…
$\pi_{N\pi}$	*yes*	*10*	*1.53e-3*