

Good vibrations: the issue of optimizing dynamical reservoirs

Workshop on ESNs / LSMs, NIPS 2006

Herbert Jaeger

International University Bremen
(Jacobs University Bremen, as of Spring 2007)

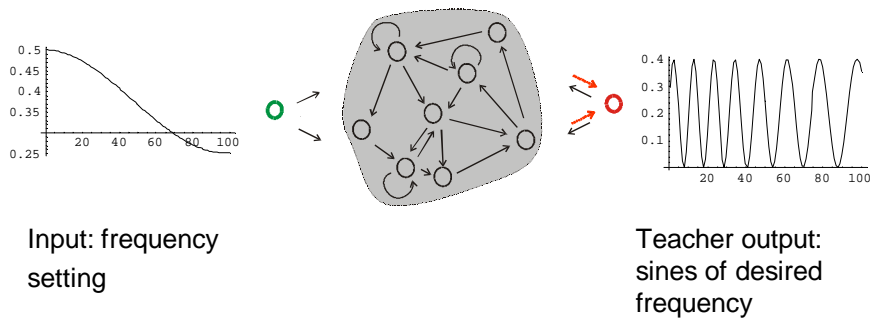


The basic idea: echo states in a dynamical reservoir



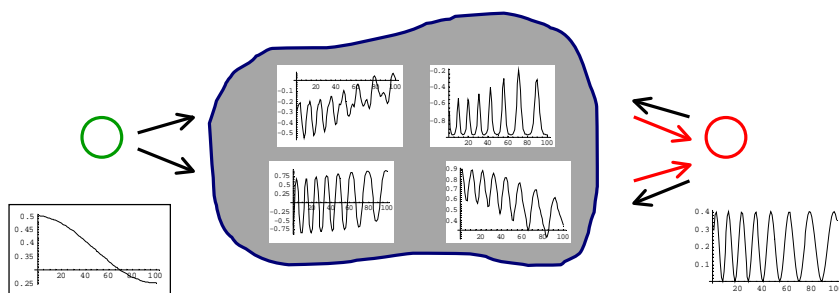
Example: a tone generator

Goal: train a network to work as a tuneable tone generator



Computing output weights

- Determine reservoir-to-output weights \rightarrow such that training output is optimally linearly combined from internal "echo" signals.



It is so simple

- A dynamical reservoir is "excited" by input signal(s), yielding a plethora of nonlinear transforms
- Combine desired output from these

It's too simple

- Quality of combined output depends on signals in the reservoir
- Works superbly on simple problems of the right kind - but does it scale?



A null version approach to optimize reservoirs



Strategy (V 0.1)

- Ultimate goal of ESN training: small test error
- Optimize reservoir size N (gives model capacity) by cross validation
- Other global control parameters (like spectral radius ρ) are optimized with a small ESN and reduced training set,
 - using some search method for minimal training error settings of global controls,
 - hoping that found settings scale to larger reservoirs.
- Thus, primary task is to optimize global "dynamics-shaping" control parameters for minimal training error



Leaky-integrator neuron ESNs¹⁾

$$\mathbf{x}(n+1) = (1-a)\mathbf{x}(n) + \tanh(s^{in}\mathbf{W}^{in}\mathbf{u}(n+1) + \rho\mathbf{W}\mathbf{x}(n) + s^{fb}\mathbf{W}^{fb}\mathbf{y}(n) + s^v\mathbf{v}(n+1))$$
$$\mathbf{y}(n+1) = g(\mathbf{W}^{out}\mathbf{x}(n+1))$$

Five global control parameters

- N reservoir size
- a leaking rate ($0 \leq a < 1$; if $a = 1$ a standard ESN results)
- s^v scaling of state noise
- s^{in} input scaling
- s^{fb} feedback scaling
- ρ spectral radius of reservoir weight matrix

($\rho < a$ necessary condition for echo state property)

1) Jaeger et al. Echo State Networks with Leaky Integrator Neurons, and Optimization of Their Global Control Parameters. Neural Networks, to appear



Demo task: recalling a structured signal



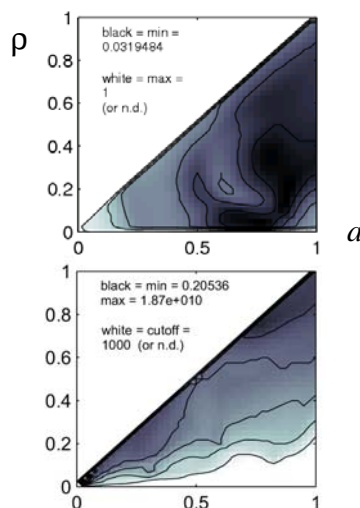
Input $u(n)$:
sum of incommensurate sines

Desired output $d(n)$:
 $u(n - 5)$

Fixed: $N = 10$, $s^v = 0$, $s^{fb} = 0$, $s^{in} = 0.3$

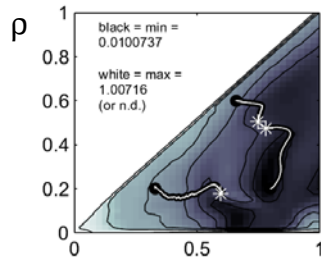
Optimized: leaking rate a , spectral radius ρ .

Echos, repercussions, reverberations, resonances...



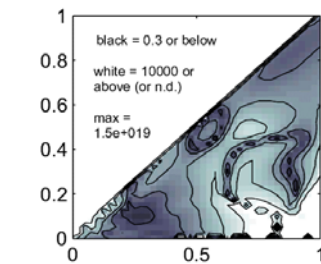
- Grey shades show log training NRMSE
- Echo state property only for $\rho < a$
- Multiple local NRMSE minima
- Explanation: ESN exploits "resonances" with various sines
- Grey shades show log average absolute output weights
- Best training NRMSE is obtained for very large output weights (order of 10,000)

Optimal global controls by gradient descent, V 0.1



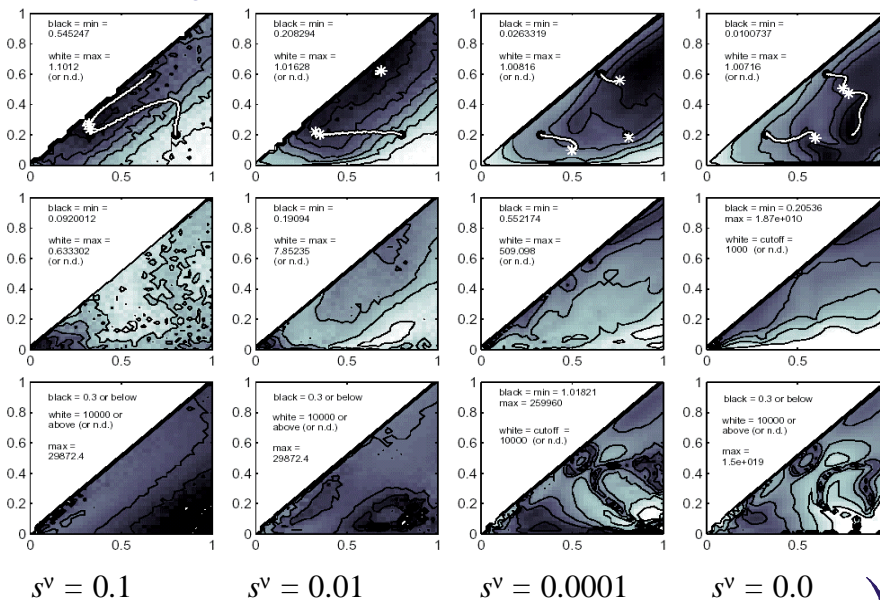
a

- Figure shows traces of stochastic gradient descent in MSE landscape over (a, ρ) -hyperplane
- 1 Mio updates, very slow motion
- Asymptotic points are clearly different from local MSE minima



- Figure shows $\partial^2 \text{MSE} / \partial a^2$
- Optimal MSE values are in areas with very large curvature (order of 10,000 and above)
- Implications for gradient descent: brittle and intolerably slow

The healing effects of state noise



Healing and concealing effects of state noise

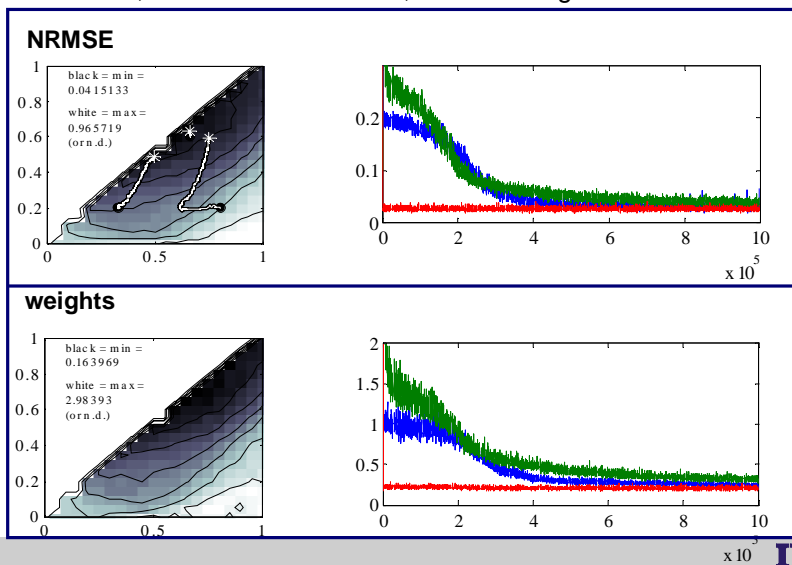
s^v	0.1	0.01	0.0001	0.0
NRMSE_{\min}	0.54	0.21	0.026	0.010
$\langle \mathbf{W}^{\text{out}} \rangle_{\max}$	0.6	7.8	510	$1.8\text{e}+10$
$\langle \partial^2 \text{MSE} / \partial a^2 \rangle$	26	107	890	$1.4\text{e}+06$
λ_{\max}	0.04	0.009	0.001	0.0000007

When adding state noise during training,

- training error increases,
- locations of optimal global controls change
- models become more resistant against perturbations
- online adaptation via LMS becomes faster (and possible at all)

A nightly observation

Same task, network size $N = 100$, noise scaling 0.01



Strategy (V 0.1), comments

Network size N and state noise size s^v : controls for statistics

- network size N controls model capacity
- state noise size s^v controls robustness and online learnability via LMS; also model capacity
- suggested: network as large as resources allow, noiselevel as large as targeted accuracy allows

Leaking rate α , spectral radius ρ , input and feedback scaling s^{fb} , s^{in} : controls for dynamics

- optimize by gradient descent if output weight size admits
- else optimize manually

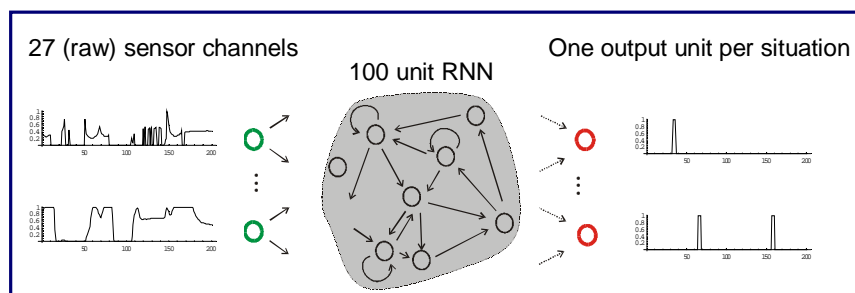
What, exactly, should we optimally optimize?

Optimization target 1: single-task test error

Tradeoff accuracy vs. robustness / online learnability

Optimization target 2: test error for multiple simultaneous tasks

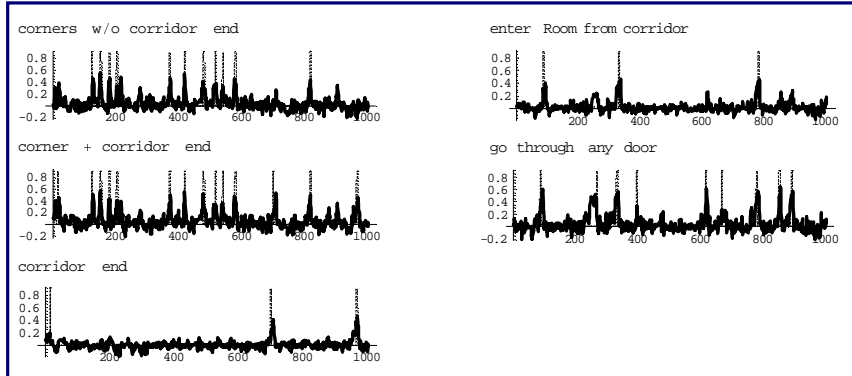
Example: situation recognition for a mobile robot¹⁾



Situation categories like "pass through door", "pass by 90° corner"

¹⁾ joint work with J.Hertzberg & F. Schönherr, Fraunhofer AIS

Results



- Works quite well.
- Could it work even better with better-tuned reservoir?

Optimizing reservoir for multiple tasks: issues

- Find criterium for combined optimality, ...
- ... and/or optimize-differentiate reservoir

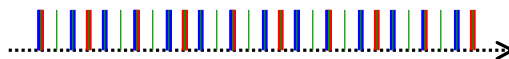
(Digression: multi-tasking ESNs)

Wanted: a simple neural network which...

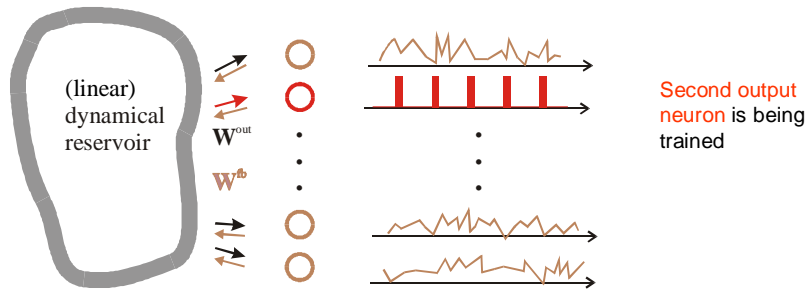
- ... can generate different "unit beats", that is, periodic unit pulses of different period length



- ... can combine (possibly phase-shifted) copies of unit beats into stable complex beat

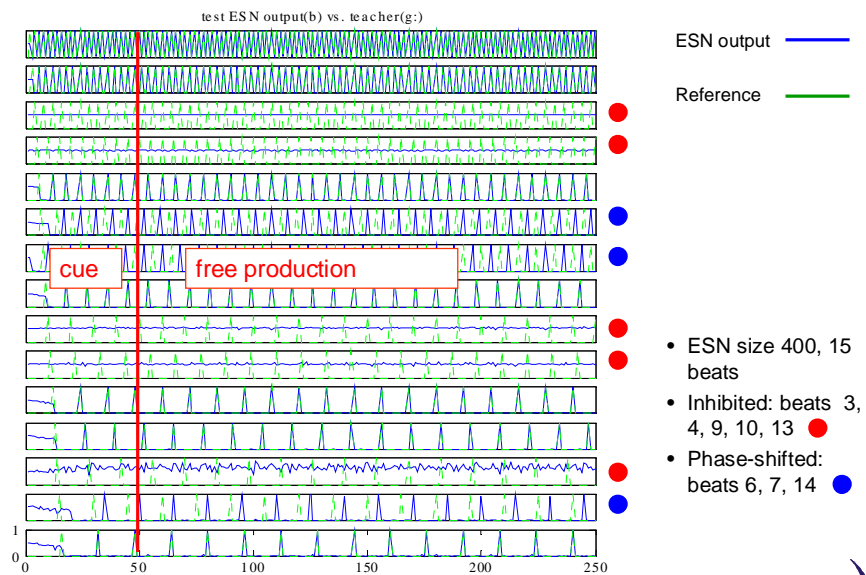


An ESN model: training setup



- Train (many) output units individually on different unit beats
- While one is trained, others feed noise into reservoir

Performance



End of digression.

Return to topic:
Optimizing reservoirs



Optimization target 3: general-purpose suitability of reservoir

- Can reservoir be optimized for a generic classes of tasks?
- For instance:
 - Tasks requiring long memory
 - Tasks requiring strong nonlinearity
 - Tasks with high-dimensional input
- Ozturk et al, "Analysis and Design of Echo State Networks for Function Approximation" (Neural Computation)
 - General general-purpose optimization through homogeneous placement of eigenvalues of reservoir weight matrix in unit disc



Optimization target 4: stability, online adaptivity with LMS, robustness, generalization

- Observation: large output weights harm all of these goals
- Math question: what's the true cause (large output weights are only a lead symptom)
- Hypothesis: the cause is a large spectral spread in reservoir signal correlation matrix
- Optimization task: can we pre-shape reservoir to have small signal crosscorrelation spectrum spread?
 - I tried at least a dozen things, no success
 - Seems really crucial for broad usability of ESNs



Summary: optimizing reservoirs

- Statistical and dynamical optimization aspects
- Optimizing target is not clear-cut:
 - Test error optimization
 - Single-task
 - Multiple-task
 - Optimization for classes of tasks
 - (General-purpose optimization?)
 - Stability, robustness, adaptivity
- Optimization is not easy
 - Multiple optima for dynamics control parameters
 - Gradient descent practically impossible when output weights are large



Thank you.