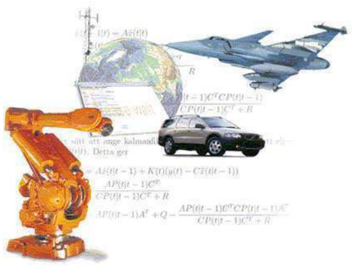
Some Classical and Some New Ideas in System Identification

Mostly Linear Models



Lennart Ljung

Reglerteknik, ISY, Linköpings Universitet

Outline

- The classic, conventional System Idenfication Setup
- Convexity Aspects
- Bias – Variance, Model Size Selection
- Regularization

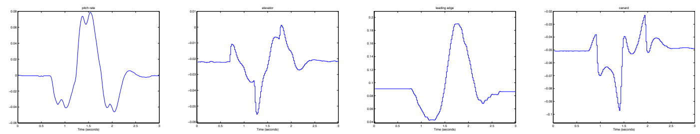
System Identification – Concrete Example

Consider a physical system, with observed input and output signals, see Figure 1. Let us take a modern military aircraft, like the Swedish fighter Gripen, as an example.

Figure : The Swedish aircraft Gripen

The Aircraft Data

From one of the earlier test flights, some data were recorded as depicted below.



(a) The output: pitch rate

(b) Control input 1: elevator angle

(c) Control input 2: leading edge flap

(d) Control input 3: canard angle

Figure : Data from an early test flight of Gripen. These data cover 3 seconds of flight and are sampled at 60 Hz.

Build a model from the data!

Try a simple difference equation relation:

$$\begin{aligned} y(t) =& a_1 y(t-1) - a_2 y(t-2) - a_3 y(t-3) \\ &+ b_{1,1} u_1(t-1) + b_{1,2} u_1(t-2) \\ &+ b_{2,1} u_2(t-1) + b_{2,2} u_2(t-2) \\ &+ b_{3,1} u_3(t-1) + b_{3,2} u_3(t-2) \end{aligned}$$

We use only the 90 first data points of the observed data. That gives certain numerical values of the 9 parameters above:

$$\begin{aligned} y(t) - 1.15y(t-1) + 0.50y(t-2) - 0.35y(t-3) \\ = -0.54u_1(t-1) + 0.04u_1(t-2) \\ +0.15u_2(t-1) + 0.16u_2(t-2) \\ +0.16u_3(t-1) + 0.07u_3(t-2) \end{aligned} \tag{1}$$

Evaluating the Model

We may note that this model is unstable – it has a pole in 1.0026, which is a correct property.
We may compare the model's 5 samples ahead predictions with the measured output (note that second half was not used for estimation):

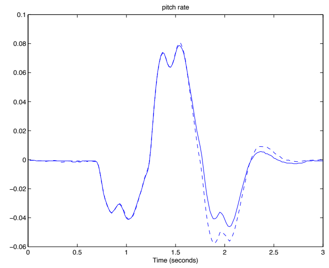


Figure : The measured output (solid line) compared to the 5 step ahead predicted one (dashed line).

Typical, Important Questions

- What type of model should be used? (like the difference equation)
- Which orders should be used? (like 3,2,2,2)
- How should the parameters be adjusted to data?
- What inputs should be applied when collecting the data?
- How to assess the quality of the estimaed model?
- How to gain confidence in the estimated model?

System Identification: State-of-the-Art Setup

A Typical Problem

Given Observed Input-Output Data: Find a Description of the System that Generated the Data [Simulator or Predictor. Linear System: Impulse response or Bode plot].

Basic Approach

Find a suitable Model Structure, Estimate its parameters, and compute the response of the resulting model

Techniques

Estimate the parameters by ML techniques/PEM (prediction error methods). Find the model structure by AIC, BIC or Cross Validation

More Formally

Models:

Model Structure: $\mathcal{M}$. Parameters: θ. Model: $\mathcal{M}(\theta)$.
Observed input–output (u, y) data up to time t: Z^t
Model described by predictor: $\mathcal{M}(\theta) : \hat{y}(t|\theta) = g(t, \theta, Z^{t-1})$.

Estimation:

log likelihood function $V_N(\theta) = \sum_{t=1}^{N} |y(t) - \hat{y}(t|\theta)|^2$
"Prediction Error Fit"
$\hat{\theta}_N = \arg\min V_N(\theta)$

Model Structure (size) determination, AIC, BIC:

$\mathcal{M}(\hat{\theta}_N) = \arg\min_{\mathcal{M},\theta} [\log V_N(\theta) + g(N)\text{dim}\theta]$
$g(N) = 2$ or $\log N$

Comment on Model Structure Selection

The model fit as measured by $\sum_{t=1}^{N} |y(t) - \hat{y}(t|\theta)|^2$ for a certain set of data will always improve as the model structure becomes larger (more parameters). The parameters will start adjusting also to the actual noise effects in the data ["Overfit"]
There are two ways of counteracting this effect:

- Compute the model on one set of (estimation) data and evaluate the fit on another (validation) data set. [Cross-Validation]
- Add a penalty term to the criterion which balances the overfit:

$$\mathcal{M}(\hat{\theta}_N) = \arg\min_{\mathcal{M},\theta} [\log V_N(\theta) + g(N)\text{dim}\theta]$$

$$AIC : g(N) = 2, \quad BIC : g(N) = \log(N)$$

AIC: Akaike's Information Criterion. BIC: Bayesian Information Criterion [= MDL: MInimum Description Length]

Model Estimate Properties

As the number of data, N, tends to infinity

- $\hat{\theta}_N \to \theta^* \sim \arg\min_\theta E|\varepsilon(t, \theta)|^2$ the best possible predictor in $\mathcal{M}$
- If $\mathcal{M}$ contains a true description of the system
  - Cov $\hat{\theta}_N = \frac{\lambda}{N}[E\psi(t)\psi^T(t)]^{-1}$ [$\psi(t) = \frac{d}{d\theta}\hat{y}(t|\theta), \lambda$: noise level]...
  - ... is the Cramér-Rao lower bound for any (unbiased) estimator.

E: Expectation. These are very nice optimal properties:

- The model structure is large enough: The ML/PEM estimated model is (asymptotically) the best possible one. Has smallest possible variance (Cramér- Rao)
- The model structure is not large enough: The ML/PEM estimate converges to the best possible approximation of the system. Smallest possible "asymptotic bias".

Experiment Design

Experiment design is the question of choosing which signal to measure, the sampling rate, and designing the input. The theory of experiment design primarily relies upon analys of how the covariance matrix
Cov $\theta_N = \frac{\lambda}{N}[E\psi(t)\psi^T(t)]^{-1}$ [$\psi(t) = \frac{d}{d\theta}\hat{y}(t|\theta)$]
depends on these variables:

$$"\min_{\mathcal{X}} \text{trace}\{C[E\psi(t)\psi^T(t)]^{-1}\}"$$

C reflecting the intended use of the model. For linear systems the input design is often expressed as selecting the spectrum (frequency contents) of u.
Bottom line: let the input's power be concentrated to frequency regions where a good model fit is essential, and where disturbances are dominating.

Model Validation and Gaining Confidence in Models

An easy method with a simple interpretation is to simulate the model with input data for which the system's response has been recorded. Then it can easily be judged how well the model can reproduce the actual system's behavior. [Cross Validation.]

This as such does not tell if all the noise free response has been covered. It is customary to check of the residuals [=measured output − (predicted) model output] have some trace of the input and/or if these prediction errors seem to be unpredictable. This is called residual analysis and there is an extensive theory for how to analyse certain correlation functions for such traces.

Linear Models

General Description

$y(t) = G(q,\theta)u(t) + H(q,\theta)e(t)$, q : shift op. e : white noise

$$G(q,\theta)u(t) = \sum_{k=1}^{\infty} g_k u(t-k), \quad H(q,\theta)e(t) = 1 + \sum_{k=1}^{\infty} h_k e(t-k)$$

Predictor

$$\hat{y}(t|\theta) = G(q,\theta)u(t) + [I - H^{-1}(q,\theta)][y(t) - G(q,\theta)u(t)]$$

Asymptotics: [Φ_u, Φ_v: Spectra of input and additive noise $v = He$.]

$$\hat{\theta}_N \to \theta^* = \arg\min_{\theta} \int_{-\pi}^{\pi} |G(e^{i\omega},\theta) - G_0(e^{i\omega})|^2 \frac{\Phi_u(\omega)}{|H(e^{i\omega},\theta)|^2} d\omega$$

$$\text{Cov}G(e^{i\omega},\hat{\theta}_N) \sim \frac{n}{N}\frac{\Phi_v(\omega)}{\Phi_u(\omega)} \text{ as } n, N \to \infty \quad n : \text{model order}$$

Common Black-Box Parameterizations:

BJ (Box-Jenkins)

$$G(q,\theta) = \frac{B(q)}{F(q)}; \quad H(q,\theta) = \frac{C(q)}{D(q)}$$
$$B(q) = b_1 q^{-1} + b_2 q^{-2} + \ldots b_{nb} q^{-nb}$$
$$F(q) = 1 + f_1 q^{-1} + \ldots + f_{nf} q^{nf}$$
$$\theta = [b_1, b_2, \ldots, f_{nf}]$$

ARX:

$$y(t) = \frac{B(q)}{A(q)}u(t) + \frac{1}{A(q)}e(t) \text{ or}$$
$$A(q)y(t) = B(q)u(t) + e(t) \text{ or}$$
$$y(t) + a_1 y(t-1) + \ldots + a_{na} y(t-na) = b_1 u(t-1) + \ldots + b_{nb} u(t-nb)$$

Common Black and Grey Parameterizations

State-Space with Possibly Physically Parameterized Matrices

$$x(t+1) = A(\theta)x(t) + B(\theta)u(t) + K(\theta)e(t)$$
$$y(t) = C(\theta)x(t) + e(t)$$

Corresponds to

$$G(q,\theta) = C(\theta)(qI - A(\theta))^{-1}B(\theta).$$
$$H(q,\theta) = C(\theta)(qI - A(\theta))^{-1}K(\theta) + I$$

Continuous Time (CT) Models

$$\dot{x}(t) = \mathcal{F}(\theta)x(t) + \mathcal{G}(\theta)u(t) + w(t)$$
$$y(t) = C(\theta)x(t) + D(\theta)u(t) + v(t)$$

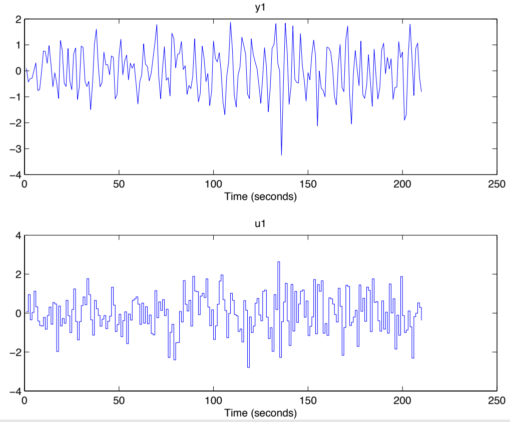
Sample it (with correct Input Intersample Behaviour):

$$x(t+1) = A(\theta)x(t) + B(\theta)u(t) + K(\theta)e(t)$$
$$y(t) = C(\theta)x(t) + e(t)$$

Now apply the discrete time formalism to this model, which is parameterized in terms of the CT parameters θ

An Example

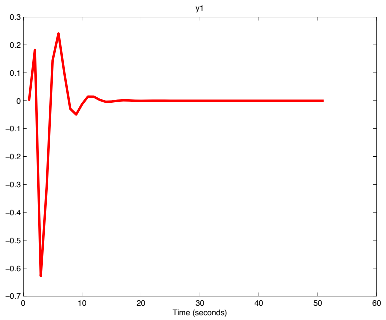
Equipped with these tools, let us now test some data (selected but not untypical). The example uses complex dynamics and few (210) data, so this is a case where asymptotic properties are not important.



Estimate a Model: State-of-the-Art

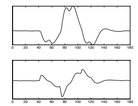
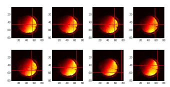
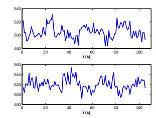
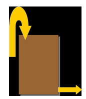
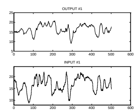
We will try the state-of-the art approach: Estimate SS models of different orders. Determine the order by the AIC criterion.

```
 for k=1:30
   m{k}= ssest(z,k);
end
(dum,n) = min(aic{:});
mss = m{n};
impulse(mss)
```



Is this a good model? An oracle tells us that the fit to the true impulse response is **83.55%** Preview: We can do better!

Status of the State-of-the-Art Framework

- Well established statistical theory
- Optimal asymptotic properties
- Efficient software
- Many applications in very diverse areas. Some examples:
  - Aircraft Dynamics:  
  - Brain Activity (fMRI):  
  - Pulp Buffer Vessel:   ;

Time-out

This is a bright and rosy picture. Any issues and problems?

- Convexity Issues: For most model structures the criterion function $V_N(\theta) = \sum_{t=1}^{N} |y(t) - \hat{y}(t|\theta)|^2$ is non-convex and multi-modal (several local minima). *Evolutionary Minimization Algorithms* could be applied, but no major successes for identification problems have been reported.
- Small data sizes – complex systems (asymptotics do not apply): Need well tuned bias–variance trade–off. Model selection rules are a bit shaky in this case.

Bias – Variance Trade Off

Any estimated model is incorrect. The errors have two sources:

- Bias: The model structure is not flexible enough to contain a correct description of the system.
- Variance: The disturbances on the measurements affect the model estimate, and cause variations when the experiment is repeated, even with the same input.

Mean Square Error (MSE) = $|\text{Bias}|^2$ + Variance.
When model flexibility ↑,Bias ↓ and Variance ↑.
To minimize MSE is a good trade-off in flexibility.
In state-of-the-art Identification, this flexibility trade-off is governed primarily by model order.

Linear Black-Box Models: Fundamental Role of ARX

ARX can Approximate Any Linear System

Arbitrary Linear System: $y(t) = G_0(q)u(t) + H_0(q)e(t)$

ARX model order n, m : $A_n(q)y(t) = B_m(q)u(t) + e(t)$

as $N >> n, m \to \infty$

$[\hat{A}_n(q)]^{-1}\hat{B}_m(q) \to G_0(q),\ [\hat{A}_n(q)]^{-1} \to H_0(q)$

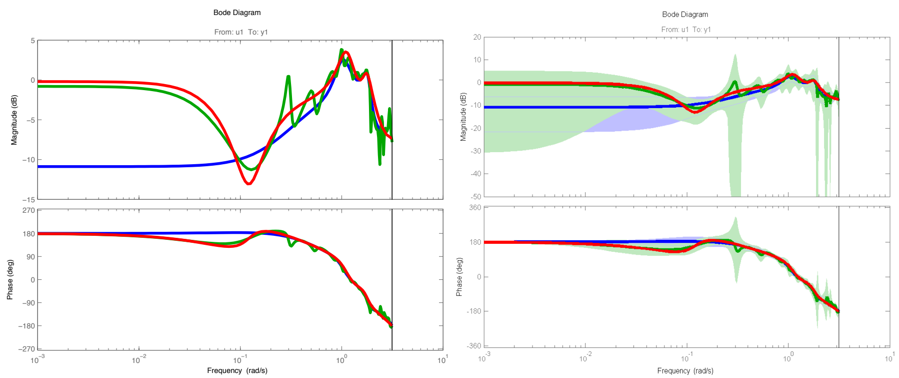
The ARX-model Is a Linear Regression

Note that the ARX-model is estimated as a linear regression $Y = \Phi\theta + E$, (Φ containing lagged y, u and θ containing a, b)
A convex estimation problem.

Virtually all methods to find a linear intial estimate for the non-convex minimization of the ML criterion are based on an ARX-model of some kind.

How High Orders are Required for ARX? Test on Our Data

Estimate ARX-model of order 10 and 30: Bode plots of models together with true system:



Order 10. Order 30. True. The high order model picks up the true curves better, but seem more ”shaky”. Look at Uncertainty regions!

How to Curb Variance/Flexibility?

The ARX approximation property is valuable, but high orders come with high variance.
Can we curb the flexibility that causes high variance other than by lower order? Regularization

High Order Models – Regularization

Curb the freedom of the model by adding a regularization term to the Least Squares Criterion:

$$Y = \Phi\theta + E$$
$$\hat{\theta}_N{}^R = \arg\min_\theta |Y - \Phi\theta|^2 + \theta^T P^{-1}\theta$$

P is the Regularization Matrix. $\hat{\theta}_N^R = (R_N + P^{-1})^{-1}\Phi^T Y$ MSE:

$$\mathsf{E}[(\hat{\theta}_N^R - \theta_0)(\hat{\theta}_N^R - \theta_0)^T] = (R_N + P^{-1})^{-1} \times$$
$$(R_N + P^{-1}\theta_0\theta_0^T P^{-1})(R_N + P^{-1})^{-1} \qquad R_N = \Phi\Phi^T,\ \theta_0 = \text{true par}$$

Minimized by $P = \theta_0\theta_0^T$: MSE = $(R_N + P^{-1})^{-1}$ How to select P?

Regularization – Bayesian Interpretation

Suppose θ is a random variable, that *a priori* (before the measurement data have been observed) is assumed to be Gaussian with zero mean and covariance matrix P: $\theta^{prior} \in N(0, P)$

$Y = \Phi\theta + E$, so Y and θ are dependent variables. After Y has been measured, we know more about θ:

$$\theta^{post} \in N(\hat{\theta}_N^R, P^{post})$$

where $\hat{\theta}_N^R$ is the regularized LS estimate from the previous slide.

So, the *a posteriori* estimate is equal to the regularized LS estimate with P as the regularization matrix.

So that is a natural way to think of a good regularization matrix: Let it mimic what is known or assumed about the parameter to be estimated. – It is the covariance matrix of the parameter vector.

Tuning the Regularization Matrix

θ is a Gaussian random vector with zero mean and covariance matrix P: $\theta \in N(0, P)$. The measured data in Φ is a known matrix, and the noise $E \in N(0, I)$. Then the output $Y = \Phi\theta + E$ is itself a Gaussian vector:

$$Y = \Phi\theta + E \in N(0, Z(P)), \quad Z(P) = \Phi P \Phi^T + I$$

So we know the pdf of Y given P, and P can be estimated by ML:

ML Estimate of P

$$\hat{P} = \arg\min_P Y^T Z(P)^{-1} Y + \log\det Z(P)$$

If P is parameterized by some hyperparameters α, $P(\alpha)$, these can be estimated by

ML Estimate of Hyperparameters

$$\hat{\alpha} = \arg\min_\alpha Y^T Z(P(\alpha))^{-1} Y + \log\det Z(P(\alpha))$$

ARX Model Priors

When estimating an ARX-model, we can think of the predictor

$$\hat{y}(t|\theta) = (1 - A(q))y(t) + B(q)u(t)$$

as made up of two impulse responses, A and B. The vector θ should thus mimic two impulse responses, both typically exponentially decaying and smooth.We can thus have a reasonable prior for θ:

$$P(\alpha_1, \alpha_2) = \begin{bmatrix} P^A(\alpha_1) & 0 \\ 0 & P^B(\alpha_2) \end{bmatrix} \qquad \text{Block Diagonal } A\&B$$

where the hyperparameters α describe decay and smoothness of the impulse responses. Typical choice:

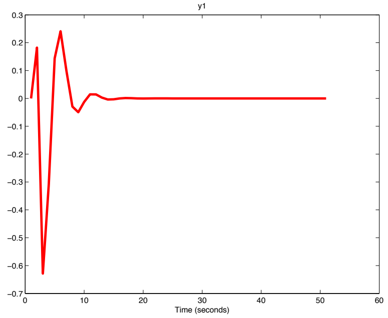
TC kernel

$P_{k,\ell} = C \min(\lambda^k, \lambda^\ell); \quad \alpha = [C, \lambda],$
$E|b_k|^2 = C\lambda^k$, $\text{corr}(b_k, b_{k+1}) = \sqrt{\lambda}$

Our Test Data: State-of-the-Art

Recall: The state-of-the art approach: Estimate SS models of different orders. Determine the order by the AIC criterion.

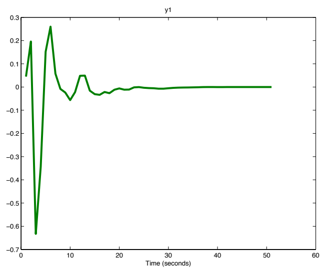
```
 for k=1:30
   m{k}= ssest(z,k);
end
(dum,n) = min(aic(:});
mss = m{n};
impulse(mss)
```



Estimate a Model: Regularized ARX

Now, let us try an ARX model with na=5, nb=60. Estimate a regularization matrix with the 'TC' kernel (2 parameters, C, λ each for the A and B parts):

```
 aopt = arxOptions;
(L,R) = arxRegul(z,[5 60 0],'TC');
aopt.Regularization.R = R;
aopt.Regularization.Lambda = L;
mr = arx(z,[5 60 0],aopt);
impulse(mr)
```

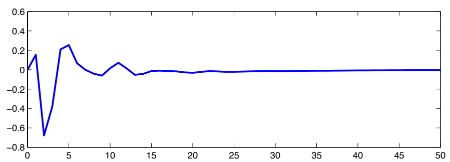


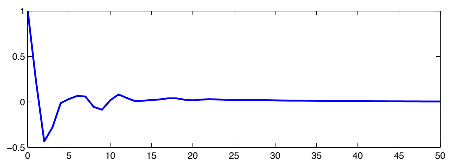
The Oracle

The examined data were obtained from a randomly generated model of order 30:

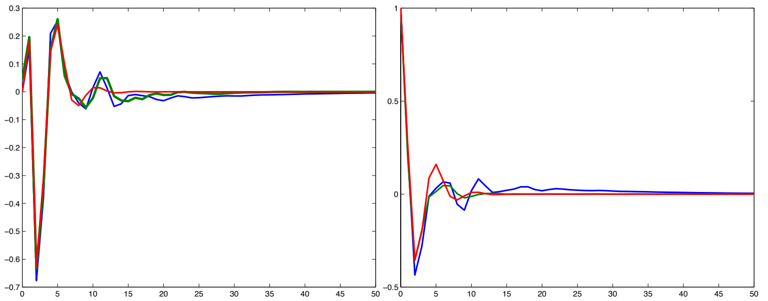
$$y(t) = G_0(q)u(t) + H_0(q)e(t)$$

The input is Gaussian white noise with variance 1, and e is white noise with variance 0.1. The impulse responses of G and H are shown at the right.





How Well Did Our Models mss And mr Do?



G : fit: **mss: 79.42% mr: 83.55%** H: fit **mss: 77.05%**, **mr: 81.59%**

Objections?

- We were just unlucky to pick order 3 (AIC). Other model selection criteria would have given better results.
  - If we ask the oracle what is the best possible state-space order for ML estimated model, the answer is **order 12 for G with a fit 82.95 %** and **order 3 for H with a fit 77.04%** So the regularized ARX -model gives better fit to both G and H than is at all possible for ML estimated state-space models [for these data].
- The R-ARX model is of order 60, and it is unfair to compare it with SS models of low order.
  - Try `mred = balred(mr,7)` to create a 7th order SS-model. It still outperforms the oracle-selected ML SS models.

Lennart Ljung
System Identification Overview ICML WS Atlanta June 20, 2013
AUTOMATIC CONTROL
REGLERTEKNIK
LINKÖPINGS UNIVERSITET

Discussion

- In this case Regularized ARX gave a much better and more flexible bias–variance trade off through the continuously adjustable hyperparameters in the regularization matrix — Compared to the state-of-the art bias–variance trade off in terms of discrete model orders.
- Can we forget about `ssest` and move over to regularized ARX?
  - No, recall that the studied situation had quite few data, and the good asymptotic properties of ML were not so prominent.
  - But one should be equipped with regularized ARX in one's toolbox
- Regularized ARX (possible followed by `balred`) can be seen as a convexification of the state-of-the art SS model estimation techniques.
  NB: Tuning of hyperparameters normally non-convex

Lennart Ljung
System Identification Overview ICML WS Atlanta June 20, 2013
AUTOMATIC CONTROL
REGLERTEKNIK
LINKÖPINGS UNIVERSITET

Conclusions

- The State-of-the art system identification relies upon a solid statistical ground, with (ML-like) parameter estimation in chosen model stuctures.
- The theory, practice, algorithms, software and applications are well established
- The non-convexity of the criterion in state-of-the-art system identification is a source of concern
- The bias-variance trade-off in terms of model order could be unsatisfactory, esp. for smaller data sets.
- Regularized ARX-models offer a fined tuned choice for efficient bias–variance trade-off and form a viable convex alternative to state-of-the-art ML techniques for linear black-box models.

Lennart Ljung
System Identification Overview ICML WS Atlanta June 20, 2013
AUTOMATIC CONTROL
REGLERTEKNIK
LINKÖPINGS UNIVERSITET