High Precision Variational Bayesian Inference of Sparse Linear Networks *

Junyang Jin a, Ye Yuan b,*, and Jorge Gonçalves a

a The Luxembourg Centre for Systems Biomedicine, avenue du Swing, 4367 Belvaux, Luxembourg
b School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, 430074, Hubei, China

Abstract
Sparse networks can be found in a wide range of applications, such as biological and communication networks. Inference of such networks from data has been receiving considerable attention lately, mainly driven by the need to understand and control internal working mechanisms. However, while most available methods have been successful at predicting many correct links, they also tend to infer many incorrect links. Precision is the ratio between the number of correctly inferred links and all inferred links, and should ideally be close to 100%. For example, 50% precision means that half of inferred links are incorrect, and there is only a 50% chance of picking a correct one. In contrast, this paper infers links of discrete-time linear networks with very high precision, based on variational Bayesian inference and Gaussian processes. Our method can handle limited datasets, does not require full-state measurements and effectively promotes both system stability and network sparsity. On several of examples, Monte Carlo simulations illustrate that our method consistently has 100% or nearly 100% precision, even in the presence of noise and hidden nodes, outperforming several state-of-the-art methods. The method should be applicable to a wide range of network inference contexts, including biological networks and power systems.

Key words: System Identification; Variational Inference; Dynamical Structure Function; Network Inference; Sparse Networks

1 INTRODUCTION
In systems biology, mathematical modelling has been central to the study of biological networks. Dynamical models are frequently developed to predict the behaviour of systems in response to external or internal stimulus for example, drug treatment or mutation. Yet only input-output dynamics are learned without exploring the topology, whereas in many other applications, comprehensive knowledge of the network topology is required. For example, the information about the structure of control systems is essential for fault diagnosis. Hence, both the inference of system dynamics and the detection of network topology are important.

Sparsity and stability are fundamental properties of most real-world networks. Communication networks, as artificial systems, are designed to be stable for robust operation and sparse to reduce energy consumption. Thus, sparsity and stability are primary constraints in network inference, especially in the case of limited data source or high amount of noise. When dealing with practical networks, often, not all the nodes in the network can be measured, because of either high experimental cost or technical limitations. The difficulty here is that many inference methods nonetheless assume full-state measurements. It is important to reconsider this issue carefully.

In recent years, kernel-based methods have become popular in the system identification community [13]. For linear systems, the methods effectively impose system stability and greatly simplify the estimation of model complexity. Kernel-based methods have successfully identified SISO continuous linear time invariant (LTI) models [13] and been further extended to discrete LTI systems [4, 5, 13]. In particular, kernel-based methods have been used to infer sparse networks described by Granger causality [6]. They have been further developed to infer the so-called sparse plus low rank networks where it is assumed that the majority of nodes can be described by a few other components that are not accessible for observation [19]. Nevertheless, kernel-based methods are not ideal for topology detection. Due to local optimal solutions, it is very difficult to achieve 100% precision.

Variational inference (VI), as empirical Bayes, is a Bayesian deterministic approximation technique that has been applied to a number of cases, including sparse regression models [1] and neural networks [2]. With well-posed models of exponential families, VI is as efficient computationally as empirical Bayes [9]. Whilst rigorous evaluation remains elusive, Monte Carlo simulations show that VI can be more accurate than empirical Bayes [9, 10]. More importantly, VI is able to estimate model evidence for each possible model structure: this

* For correspondence: yye@hust.edu.cn.
This paper combines Gaussian processes and VI to infer sparse networks\(^1\). Dynamical structure functions are used to describe sparse linear networks where the information of hidden nodes is encoded via transfer functions. VI is employed to identify system dynamics and infer network topology. Moreover, by applying backward selection strategies, the proposed method encourages high inference precision. Monte Carlo simulations show that the network is a causal system \([16]\). Matrix inference precision so that almost all inferred links are correct.

The paper is organized as follows. Section 2 introduces dynamical structure function and formulates the full Bayesian model. Section 3 discusses network inference using variational inference and analyses algorithm properties. Section 4 compares the method with other approaches via Monte Carlo simulations. Finally, Section 5 concludes and discusses further developments in this field.

2 MODEL FORMULATION

2.1 The dynamical structure function

The sparse network of \(n\) nodes is described by a linear state space model as follows:

\[
\begin{align*}
  x(t + 1) &= Ax(t) + B_u u(t) + B_e e(t) \\
  y(t) &= C x(t),
\end{align*}
\]

where \(x \in \mathbb{R}^n\) are states of the system, each of which represents a node. \(u \in \mathbb{R}^m\) denote inputs. \(y \in \mathbb{R}^p\) present the measurements of the states. \(e \in \mathbb{R}^q\) is i.i.d white Gaussian noise with zero mean and covariance matrix \(P_e\). Without loss of generality, \(P_e\) is assumed to be diagonal. \(A \in \mathbb{R}^{n \times n}\), \(B_u \in \mathbb{R}^{n \times m}\), \(B_e \in \mathbb{R}^{n \times q}\) and \(C \in \mathbb{R}^{n \times p}\) are system matrices.

It is normal in practice that some of the nodes are unobservable (hidden states). Assume the first \(p < n\) states are observable (i.e. \(C = [I, 0]\)). To avoid inferring hidden states, they are removed from the model. Dynamical structure functions (DSF) encode the information of hidden states via transfer functions \([17]\):

\[
Y = QY + PU + HE,
\]

where \(Q\), \(P\) and \(H\) are transfer matrices, each element of which is a strictly proper transfer function, indicating that the network is a causal system \([16]\). Matrix \(Q\) implies the connectivity among observable nodes, whose diagonal elements are zero. \(P\) and \(H\) matrices relate inputs and process noise to nodes, respectively. The topology of the network (i.e. model structure) is reflected by the zero structure of these three matrices. For example, if \([Q]_{ij} = 0\), the \(j\)th node does not control the \(i\)th node. Model structures are denoted by \(M_0\) and \(M_p\) presents the number of links. In particular, \(M_0\) represents the fully-connected topology. The internal dynamics of the network are described by the transfer functions. The order of a transfer function is relevant to the number of hidden states involved in that regulation pathway.

The input-output map of the network is associated with the DSF as \(Y = G_u U + G_e E\) where \(G_u = (I - Q)^{-1}P\) and \(G_e = (I - Q)^{-1}H\). Ideally, the input-output map can be perfectly recovered based on measurement data. Nevertheless, the corresponding DSF may not be unique, meaning that the network topology is unidentifiable. To ensure the inference problem is well-posed, additional constraints are imposed \([8]\). A sufficient condition for network identifiability is that matrix \(H\) is diagonal so that \(p - 1\) elements in each column of \([Q, P, H]'\) are known to be zero. In what follows, we make following assumptions so that no prior knowledge of matrix \(P\) (structure of input channels) is required.

**Assumption 1** Noise matrix \(H\) is diagonal, monic \((\lim_{q \to \infty} qH = I)\) and minimal phase.

Stability and sparsity are the basic nature of many practical networks such as biological networks and power systems. Therefore, we assume the target network is stable and sparse.

**Assumption 2** Each elements of transfer matrices, \(Q\) and \(P\) are stable. Matrices \(Q\) and \(P\) are sparse.

2.2 The likelihood distribution

After simple manipulations, the DSF in (2) can be rewritten as:

\[
Y = F_y Y + F_u U + \bar{E}.
\]

where

\[
F_u = (qH)^{-1} P, \quad F_y = I - (qH)^{-1} (I - Q), \quad \bar{E} = q^{-1} E.
\]

Model (3) is a valid causal system. According to the assumptions, transfer matrices, \(F_u\) and \(F_y\) are stable. In addition, since \(H\) is diagonal, \(F_u\) and \(F_y\) have the same zero structure as \(P\) and \(Q\). As a result, \(F_u\) and \(F_y\) are sparse matrices and imply the network topology.

Model (3) is expressed in a non-parametric way. By doing so, the selection of model complexity is avoided and, more importantly, system stability can be promoted effectively. The dynamical system for the \(i\)th target node, is formulated by

\[
y_i(t) = \sum_{j=1}^{p} \sum_{k=1}^{\infty} h_{ij}^p (k) y_j(t - k) + \sum_{j=1}^{m} \sum_{k=1}^{\infty} h_{ij}^m (k) u_j(t - k) + e_i(t) \text{ where } h_{ij}^p \text{ and } h_{ij}^m \text{ are the impulse responses of transfer functions } [F_{y_{ij}}^p \text{ and } F_{y_{ij}}^m] \text{ and}
\]
functions are Tuned/Correlated kernels (TC kernel). TC kernel has been widely used to characterize stable
impulse responses [3]. Other valid kernels include Diagonal/Correlated kernel (DC kernel)\(^2\) and second order
stable spline kernel (SS kernel)\(^3\) [7,13]. As a result, the prior distribution for \( w \) is:

\[
p(w|\lambda, \beta, \sigma, \mathcal{M}_k) = \prod_{q=1}^{L} \mathcal{N}(w_{q,i}|0, \sigma^{-1} K_{q,i}^{-1}),
\]

where \( \beta \) are hyperparameters of TC kernels, which control the exponential decaying rate of impulse responses.
\( \lambda \) are scale variables of the kernel functions. In the Bayesian model, \( \lambda \) influence the probability of model
structure (i.e. network topology). As \( \lambda \) approaches inﬁnity, distribution \( p(w_i|\lambda_i, \beta_i, \sigma) \) deploys an impulse at
the origin, enforcing zero impulse responses. In this case, the \( i \)th node or input does not control the target node.
As the standard setting of variational inference, noise precision \( \sigma \) is also used to scale the covariance matrix.
\( K_{q,i} \in \mathbb{R}^{T_q \times T_q} \), \( \lambda = [\lambda_1, \ldots, \lambda_M] \) , \( \beta = [\beta_1, \ldots, \beta_M] \) and

\[
[ K_{q,i} ]_{t,s} = k(t, s; \beta_i), k(t, s; \beta_i) = \beta_i^{\text{max}}(t, s) \quad 0 < \beta_i < 1, \lambda_i \geq 0.
\]

Since \( \sigma \) is non-negative, the Gamma distribution is assigned as its conjugate prior. Without speciﬁc preference on \( \sigma \), parameters \( a_0 \) and \( b_0 \) of the distribution are set to 0.001, resulting in a non-informative prior:
\( p(\sigma|a_0, b_0) = \text{Gamma}(\sigma|a_0, b_0) \). In addition, hyperpriors are assigned to hyperparameters to complete the hierarchy.
For hyperparameter \( \lambda_i \), the Gamma distribution is applied as the conjugate prior. Similar to \( \sigma \), a non-
informative prior is adopted. For hyperparameter \( \beta_i \), the uniform distribution on (0, 1) is employed as the prior,
i.e., \( p(\beta_i) = 1, 0 < \beta_i < 1 \).

2.4 The full Bayesian model

By incorporating the likelihood and prior distributions, the full Bayesian distribution is:

\[
p(w, \sigma, \lambda, \beta|Y, \mathcal{M}_k) \propto \prod_{q=1}^{L} \left\{ (2\pi\sigma^{-1})^{-\frac{N_q-T_q}{2}} \exp\{-\frac{\sigma}{2} \| Y_q - \Phi_q w_q \|_2^2 \} \right\} \times \left\{ 2\pi\sigma^{-1} \Lambda^{-1} K^{-1} \right\}^{-\frac{1}{2}} \exp\{-\frac{\sigma}{2} \| Y_q - \Phi_q w_q \|_2^2 \}
\]

\[
\times \frac{b_0^{a_0} \sigma^{a_0-1} e^{-b_0 \sigma}}{\Gamma(a_0)} \prod_{i=1}^{M_k} \frac{b_0^{a_0} \lambda_i^{a_0-1} e^{-b_0 \lambda_i}}{\Gamma(a_0)} 
\]

where \( K_q = \text{blkdiag}\{K_{q,1}, \ldots, K_{q,M_k}\} \) and \( \Lambda_q = \text{diag}\{\lambda\} \otimes I_{T_q} \).

---

\(^2\) \( k_{DC}(t, s; \beta_1, \beta_2) = \beta_1^{\text{max}}(t) \beta_2^{\text{max}}(s), \beta_1 \in (0, 1) \) and \( \beta_2 \in (-1, 1) \)

\(^3\) \( k_{SS}(s, t; \beta) = \beta^{\text{max}}(s) + \beta^{\text{max}}(t), \beta \in (0, 1) \)
3 VARIATIONAL INFERENCE OF DYNAMICAL STRUCTURE FUNCTIONS

3.1 Update of random quantities

For each model structure \( M_k \), the corresponding full Bayesian model, \( p(w, \sigma, \lambda, \beta|Y, M_k) \) is approximated by a candidate distribution, \( Q(w, \sigma, \lambda, \beta|M_k) \) using the mean field factorization:

\[
Q(w, \sigma, \lambda, \beta|M_k) = q(w, \sigma|M_k)q(\lambda|M_k)q(\beta|M_k),
\]

(10)

where \( Q(\cdot) \) and \( q(\cdot) \) are valid probability distributions. Hereafter, the symbol, \( M_k \) is suppressed to simplify the notation.

The factors of (10) are formulated in sequence. In what follows, the terms independent on the random variables of the factor under consideration are ignored for convenience. To begin with, factor \( q(w, \sigma) \) is solved as the Gaussian-Gamma distribution:

\[
\ln q(w_q, \sigma) = \ln \mathcal{N}(w_q|\mu_q, \sigma^{-1}\Sigma_q) - \ln \text{Gamma}(\sigma|\alpha^w, \beta^w),
\]

(11)

where

\[
\Sigma_q^{-1} = \Phi_q'\Phi_q + E_\lambda(\Lambda_q)E_\beta(K_q^{-1}), \quad \mu_q = \Sigma_q\Phi_q'Y_q
\]

\[
a^w = \sum_{q=1}^L N_q - T_q + a_0,
\]

\[
b^w = b_0 + \frac{1}{2} \sum_{q=1}^L (Y_q'Y_q - \mu_q'\Sigma_q^{-1}\mu_q),
\]

\[
E_{\sigma, w}(\sigma w_q w_q') = \frac{a^w}{b^w} \mu_q \mu'_q + \Sigma_q, \quad E_{\sigma, w}(\sigma w_q) = \frac{a^w}{b^w} \mu_q.
\]

Following the same procedure, factor \( q(\lambda) \) is solved as independent Gamma distributions:

\[
\ln q(\lambda_i) = \ln \text{Gamma}(\lambda_i|a_{\lambda_i}, b_{\lambda_i}),
\]

(13)

where

\[
a_{\lambda_i} = \sum_{q=1}^L T_q + a_0, \quad E_\lambda(\Lambda_i) = \frac{a_{\lambda_i}}{b_{\lambda_i}},
\]

\[
b_{\lambda_i} = b_0 + \frac{1}{2} \text{trace} \left[ \sum_{q=1}^L E_\beta(K_q^{-1}) E_{\sigma, w}(\sigma w_q w_q')T_q \right].
\]

Finally, factor \( q(\beta) \) is calculated. Unlike the other random variables, factor \( q(\beta) \) has no closed-form expression. Nevertheless, the elements of \( \beta \) are independently distributed as:

\[
q(\beta_i) = c_i \prod_{q=1}^L |K_{q, i}|^{-\frac{1}{2}} \exp \left\{ -\frac{1}{2} E_\lambda(\Lambda_i) \text{trace} \left[ K_{q, i}^{-1} E_{\sigma, w}(\sigma w_q w_q')T_q \right] \right\},
\]

(15)

where \( c_i \) is the unknown normalization constant.

3.2 Lower bound of model evidence

Combining all the factors above, the lower bound, \( L[Q(w, \sigma, \lambda, \beta|M_k)] \) of model evidence \( p(Y|M_k) \) is:

\[
L[Q(w, \sigma, \lambda, \beta|M_k)]
\]

\[
= \frac{1}{2} \sum_{q=1}^L \ln |\Sigma_q| - \left[ \frac{a^w}{b^w} \|Y_q - \Phi_q \mu_q\|_2^2 + \text{trace}(\Phi_q \Sigma_q \Phi'_q) \right]
\]

\[
- b^w_0 \frac{a^w}{b^w} - a^w_0 \ln b^w_0 - \sum_{i=1}^{M_k} a_{\lambda_i} \ln b_{\lambda_i} + \frac{M_k \sum_{q=1}^L T_q}{2} + M_k [a_0 \ln b_0 - \ln \Gamma(a_0)]
\]

\[
- \sum_{i=1}^{M_k} b_0 a_{\lambda_i} \ln b_{\lambda_i} - a_{\lambda_i} - \ln \Gamma(a_{\lambda_i}) + \ln c_i,
\]

(16)

where the constant terms independent on \( M_k \) are ignored.

3.3 Algorithm for variational inference

The VI algorithm is summarized in Algorithm 1. Since \( q(\beta) \) is only known up to a normalization constant, \( E(K^{-1}) \) cannot be calculated analytically. Hence, Metropolis-Hastings sampling methods are used to estimate the expectation. With \( N \) samples \( \{\beta^k\}_{k=1}^N \), \( E(K_{q, i}^{-1}) \) is estimated as \( E(K_{q, i}^{-1}) = \frac{1}{N} \sum_{k=1}^N (K_{q, i}^{-1})^{-1} \). In addition, in order to calculate the lower bound of model evidence, one has to estimate the normalization constant, \( c_i \) of \( q(\beta_i) \) in (15). Since \( \beta_i \) is a scalar on \((0, 1)\), numeric integration methods (e.g. adaptive quadrature [15]) are sufficient to give an accurate estimation given \( \int q(\beta_i) d\beta_i = 1 \). The theoretical analysis of the algorithm can be found in [11].

Model selection is based on the posterior distribution of model structures (i.e. \( p(M_k|y) \propto p(y|M_k)p(M_k) \)). Assuming equal probability for each model structure, the distribution is proportional to model evidence, \( p(M_k|y) \propto p(y|M_k) \). The by-product of VI provides a
lower bound of model evidence, which can be used to determine the most probable model structure (network topology). We come up with a heuristic strategy that applies backward selection to select model structures as shown in Algorithm 2.

Algorithm 2 Variational inference of network topology
1: Implement Algorithm 1 with model structure $M_{0}$.
2: Set the threshold: $R = \text{ase}(\sum_{q}||w_{q,1}||, \ldots, ||w_{q,n+p}||)$.
3: for $k=1:n+p-1$ do
4: \hspace{1cm} $I = \{i|\sum_{q}||w_{q,i}|| \leq R_k\}$
5: \hspace{1cm} Remove the links in index set $I$, producing $M_k$
6: \hspace{1cm} Run Algorithm 1 with $M_k$
7: \hspace{1cm} Store the value of $L(Q(w, \sigma, \lambda, \beta|M_k))$
8: end for
9: $M_{opt} = \text{argmax}_{M_k} L(Q(w, \sigma, \lambda, \beta|M_k))$

4 SIMULATION

We conducted Monte Carlo simulations and compared our method with another kernel-based system identification approach. The state-of-the-art method [6] applies empirical Bayes to estimate hyperparameters and to detect network topology, where DC (Kernel_DC), SS (Kernel_SS) and TC (Kernel_TC) kernels were all used for inference. The DSF networks for test were generated randomly with diverse topologies (including an extremely sparse type, a ring structure), simulated under various noise levels and inferred using time-series data of different lengths.

Two criteria are applied to evaluate the performance of algorithms, namely True Positive Rate (TPR) and Precision (PREC). TPR shows the percentage of the true links in the ground truths that were successfully inferred. TPR implies the information richness of the inference result. PREC equates to the rate of the correct links over the all inferred. PREC indicates the reliability or accuracy of the inferred network. Hence, ensuring a high PREC is the priority in real applications. If, for example, PREC is below 50%, one cannot tell which inferred links are correct. To investigate whether the inferred networks have internal dynamics consistent with the ground truths, the estimated models were simulated to predict the validation dataset that was not used for inference.

4.1 Random DSF networks

100 networks were generated with random topologies and internal dynamics. All networks contained 15 nodes in total. Each node was independently driven by an input that was known and process noise that was assumed to be unknown during inference. DSF models were produced from state space models (1). To be specific, a sparse stable matrix $A \in \mathbb{R}^{15 \times 15}$ was first yielded randomly using the function $\text{sprand}(n, n, \text{density})$ in Matlab. The brute force strategy was applied to guarantee that matrix $A$ was Hurwitz (that is, no eigenvalue was outside the unit circle of the complex plane) and that no isolated nodes existed in the network.

To simulate the models, inputs and process noise were both i.i.d. white Gaussian signals. The variance of inputs was fixed to 1 whilst that of process noise varied. The Signal-to-Noise ratio is defined as $SNR = 10 \log_{10}(\frac{\sigma_u}{\sigma_e})$ where $\sigma_u$ and $\sigma_e$ are signal variance of inputs and noise, respectively. The first 10 states of the models were measured, leaving the rest 5 as hidden nodes. The length of impulse responses after truncation was set to 20. The data for inference were collected with different lengths.

The average TPR and PREC over 100 trials are recorded in Table 1-3. In the best-case scenario (no noise), VI outperforms all the other methods. In particular, almost all the inferred links of VI are correct ($\text{PREC} \approx 100\%$), regardless of the number of data points. Meanwhile, VI is able to identify most true links of the ground truths.

<table>
<thead>
<tr>
<th>No Noise</th>
</tr>
</thead>
<tbody>
<tr>
<td>45</td>
</tr>
<tr>
<td>PREC</td>
</tr>
<tr>
<td>Kernel_DC</td>
</tr>
<tr>
<td>Kernel_SS</td>
</tr>
<tr>
<td>Kernel_TC</td>
</tr>
<tr>
<td>VI</td>
</tr>
</tbody>
</table>

4.2 Ring networks

100 networks with the fixed ring structure were generated and simulated following the same protocol in the last section. Each node was driven by independent process noise. Only one input entered the network through a
5 CONCLUSION AND DISCUSSION

This paper has applied variational inference to identify DSF models given measured time series data. No prior knowledge of the hidden nodes including their number and internal connectivity is required. Both the system dynamics and topology of sparse linear networks can be inferred accurately. The proposed method achieves this by incorporating kernel-based methods to promote system stability and by introducing VI to imposing network sparsity. Monte Carlo simulations imply that our method always produces reliable inference result regardless of challenging experimental conditions (for example, lower number of data points, high levels of noise, and extremely sparse topologies). The inference of links is highly accurate (with nearly 100% confidence): only a few true links are missed. Therefore, the developed method appears highly reliable for real-world applications.

Overall, the value of this approach is that it outperforms kernel-based methods at least in regard to TC kernel. Our method raises the reliability of inference results to the highest level (100% PREC). In particular, our method is applicable to real-world networks where full state measurements are normally unavailable such as gene regulatory networks.

Future developments should take nonlinearity into account. Considering most real-word networks are nonlinear, it is necessary to extend our framework to black-box nonlinear systems.

Table 2

Inference of random networks with 10dB SNR

<table>
<thead>
<tr>
<th></th>
<th>10dB</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>200</td>
<td>300</td>
</tr>
<tr>
<td></td>
<td>PREC</td>
<td>TPR</td>
</tr>
<tr>
<td>Kernel_DC</td>
<td>95.6</td>
<td>77.1</td>
</tr>
<tr>
<td>Kernel_SS</td>
<td>74.1</td>
<td>82.5</td>
</tr>
<tr>
<td>Kernel_TC</td>
<td>92.5</td>
<td>46.5</td>
</tr>
<tr>
<td>VI</td>
<td>100</td>
<td>68.2</td>
</tr>
</tbody>
</table>

Table 3

Inference of random networks with pure noise

<table>
<thead>
<tr>
<th></th>
<th>No Input</th>
<th>300</th>
<th>1000</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>PREC</td>
<td>TPR</td>
<td>PREC</td>
</tr>
<tr>
<td>Kernel_DC</td>
<td>77.6</td>
<td>72.9</td>
<td>80.1</td>
</tr>
<tr>
<td>Kernel_SS</td>
<td>64.3</td>
<td>69.6</td>
<td>81.7</td>
</tr>
<tr>
<td>Kernel_TC</td>
<td>77.9</td>
<td>60.3</td>
<td>87</td>
</tr>
<tr>
<td>VI</td>
<td>100</td>
<td>56.5</td>
<td>100</td>
</tr>
</tbody>
</table>

Table 4

Inference of ring networks with 10dB SNR

<table>
<thead>
<tr>
<th></th>
<th>10dB</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>200</td>
<td>300</td>
</tr>
<tr>
<td></td>
<td>PREC</td>
<td>TPR</td>
</tr>
<tr>
<td>Kernel_DC</td>
<td>51.9</td>
<td>75.4</td>
</tr>
<tr>
<td>Kernel_SS</td>
<td>42.2</td>
<td>76.5</td>
</tr>
<tr>
<td>Kernel_TC</td>
<td>76.9</td>
<td>19.3</td>
</tr>
<tr>
<td>VI</td>
<td>100</td>
<td>27.5</td>
</tr>
</tbody>
</table>

References