Adjoint Operator Methods for Efficient Functional Reliability

A. John Arul
IGCAR,India.
Work done in collaboration with
Prof. Kannan Iyer, IITB Mumbai

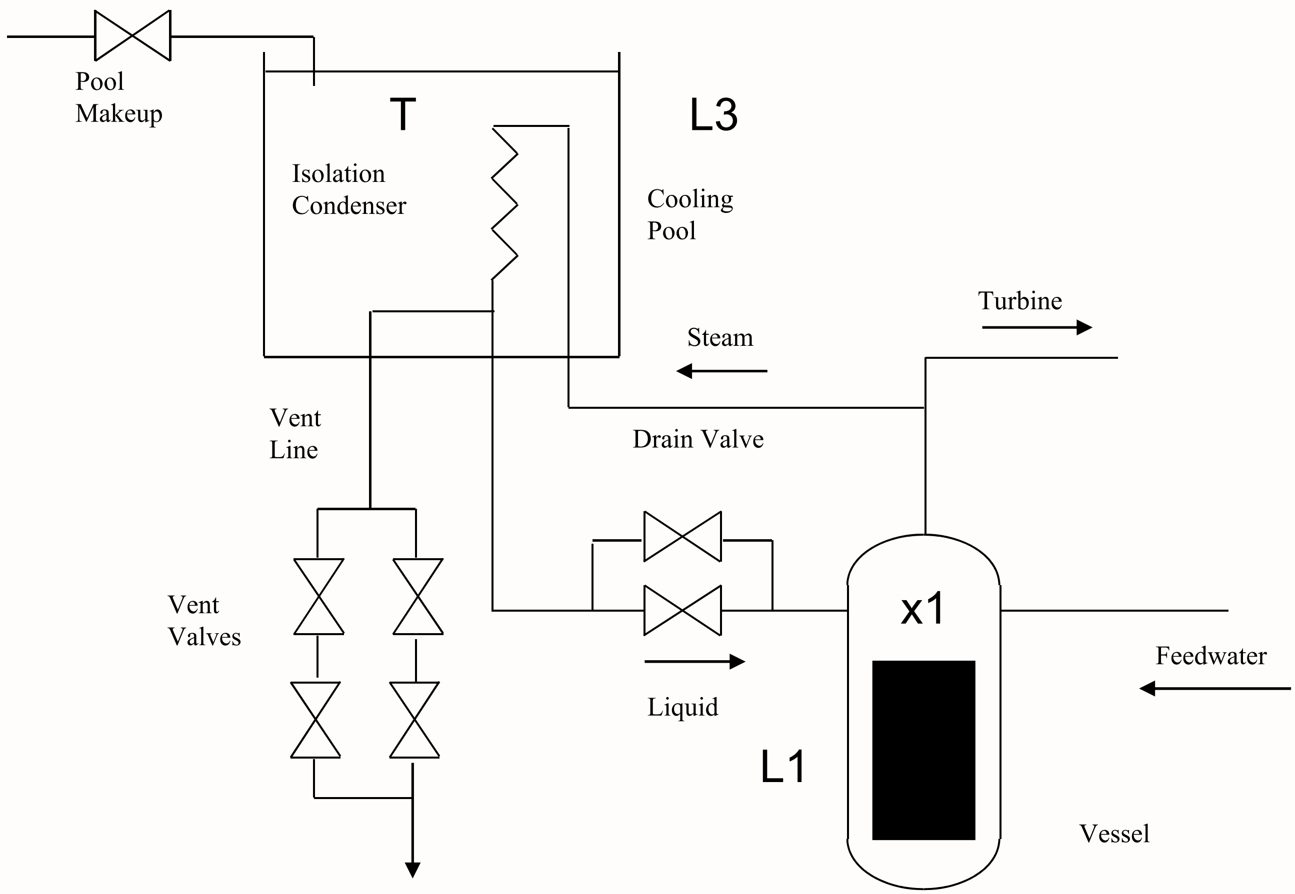
PRESENTATION OUTLINE

- Passive versus Active Systems
- The Problem of Functional Reliability
- Major steps in Functional Reliability Assessment
- Approaches for Efficiency
- Adjoint Operators
- Continuous Adjoint -Simple Application
- Automatic Differentiation
- Application - Tools
- Conclusion

Passive System Definition

- No external energy requirement
  - No AC power requirement (Grid power, DG Power)
- No moving mechanical parts
  - Like pumps, valves.
- No signals
  - Analogue or digital signal processing for sensing, control.
- Examples: Structures, Thermo-siphon based heat transport, gravity driven coolant injection.

Scheme of Isolation Condenser for a SBWR



Classification of Passive System

- **Category A**
  - No AC power, No Moving working fluid
  - No Moving parts, No signals
- **Category B**
  - No AC power, Moving working fluid
  - No Moving parts, No signals
- **Category C**
  - No AC power, Moving working fluid
  - Moving parts, No signals
- **Category D**
  - No AC power, Moving working fluid
  - Moving parts, signals

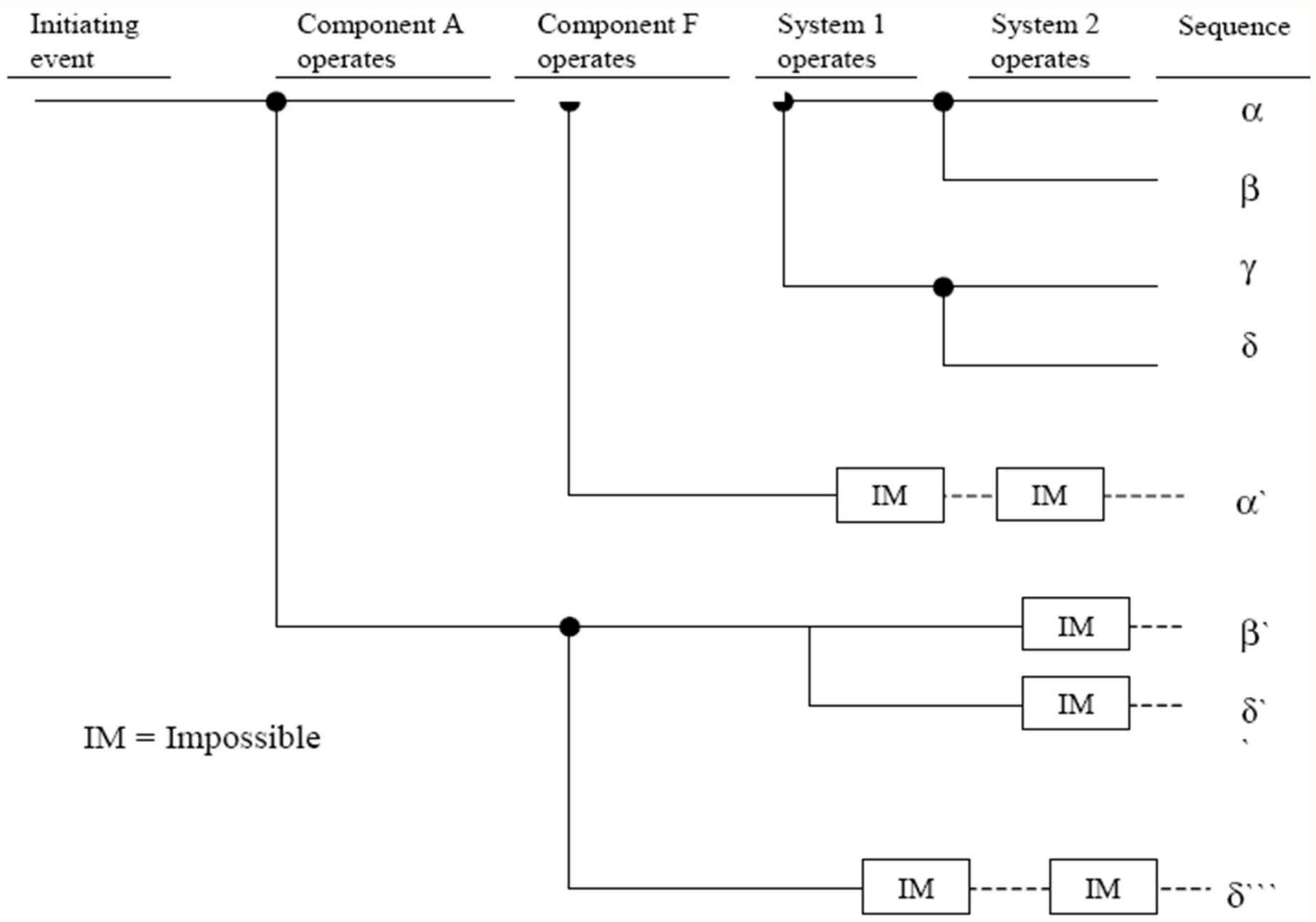
Safety System Reliability Requirement: Active/Passive

Nuclear Reactor Heat Transport applications- Regulatory requirement for (un)reliability ~ 1E-7/de.

Active System-Reliability:

- If pumps are essential, difficult to achieve unreliability < 1E-5 /de (1E-5 /h *10h *0.1)
- Reliability Analysis Method/Tools:
- FT/ET → System structure function → Boolean algebra +data→ Reliability.
- System reliability is obtained mainly as combination of active component (pump/valves) reliability.
- Functional failure is expected to be relatively small and neglected.

Event Tree


Initiating event
Component A operates
Component F operates
System 1 operates
System 2 operates
Sequence
α
β
γ
δ
IM
IM
α`
IM
β`
IM
δ`
IM
IM
δ```
IM = Impossible

Fault Trees

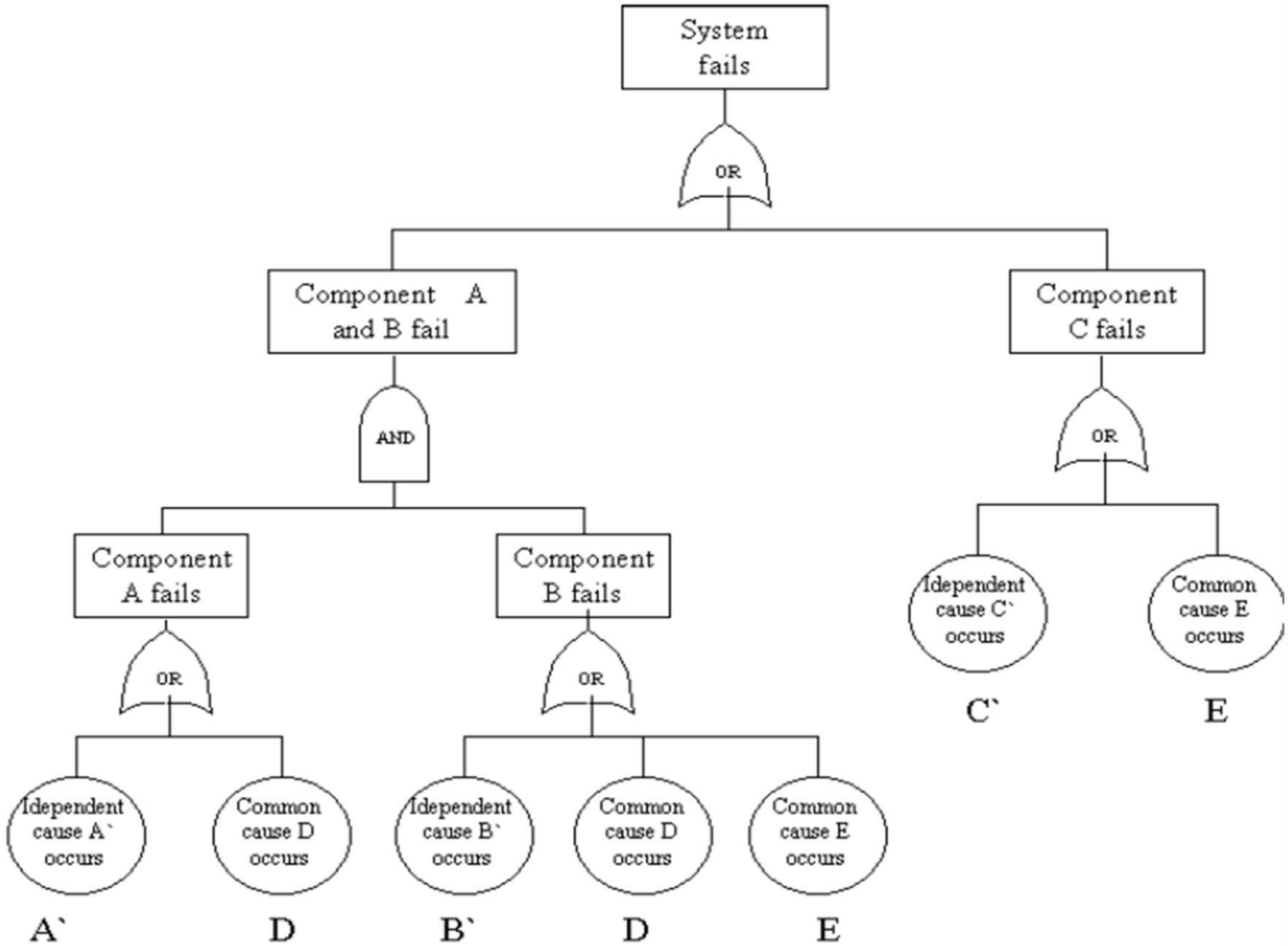


Figure 2 Fault tree for a three-component system with independent and common causes

PFBR Shutdown System

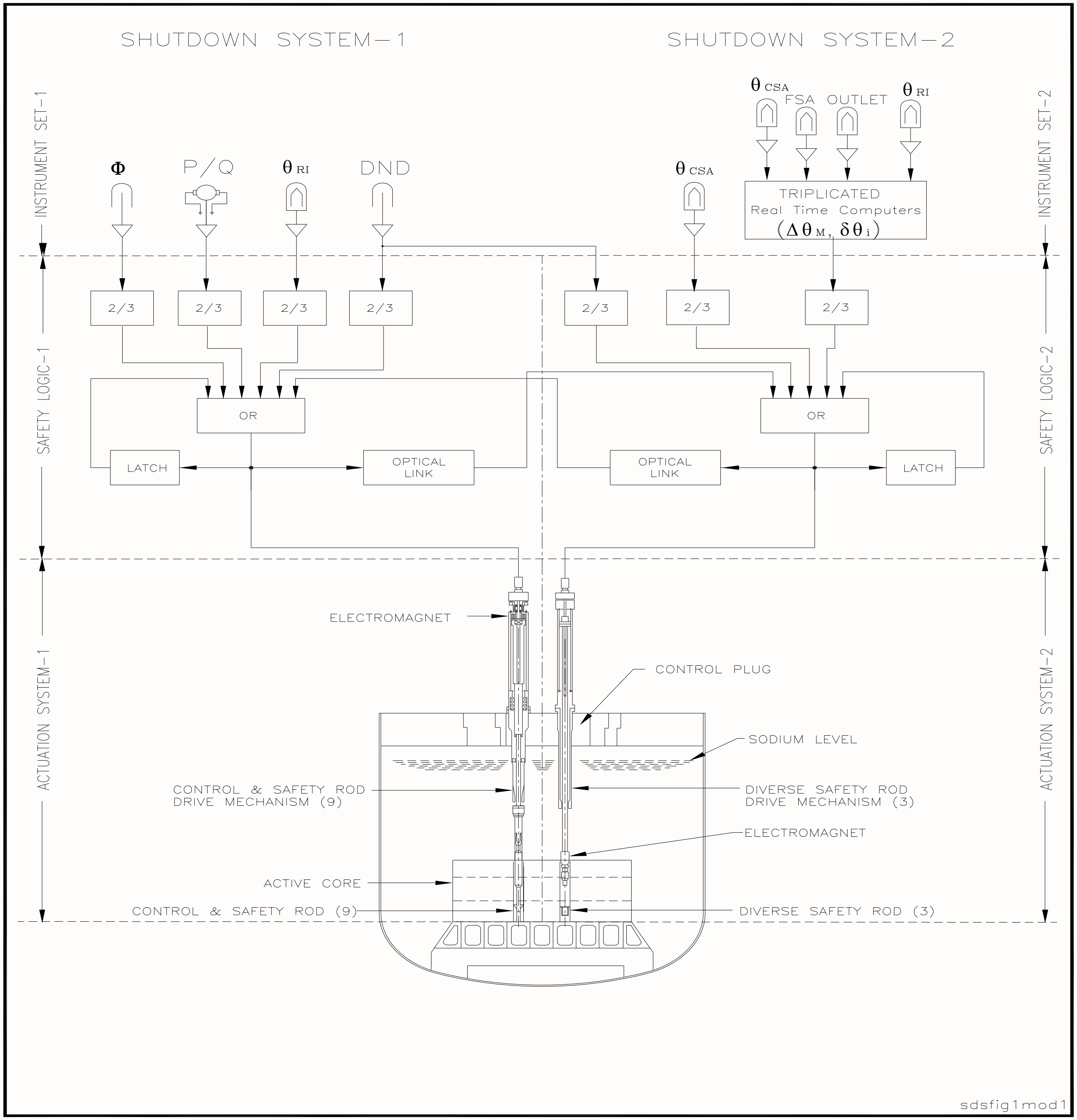
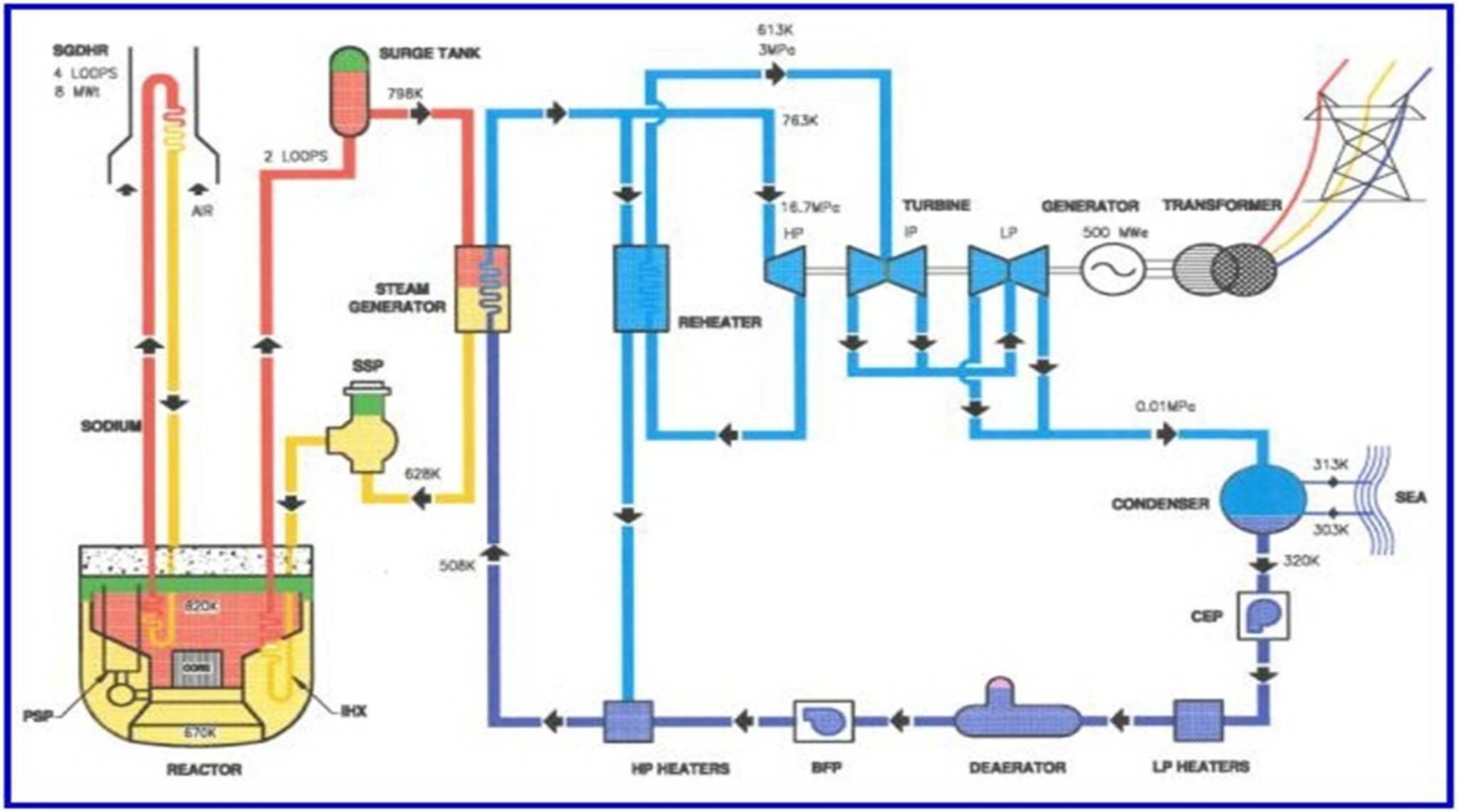


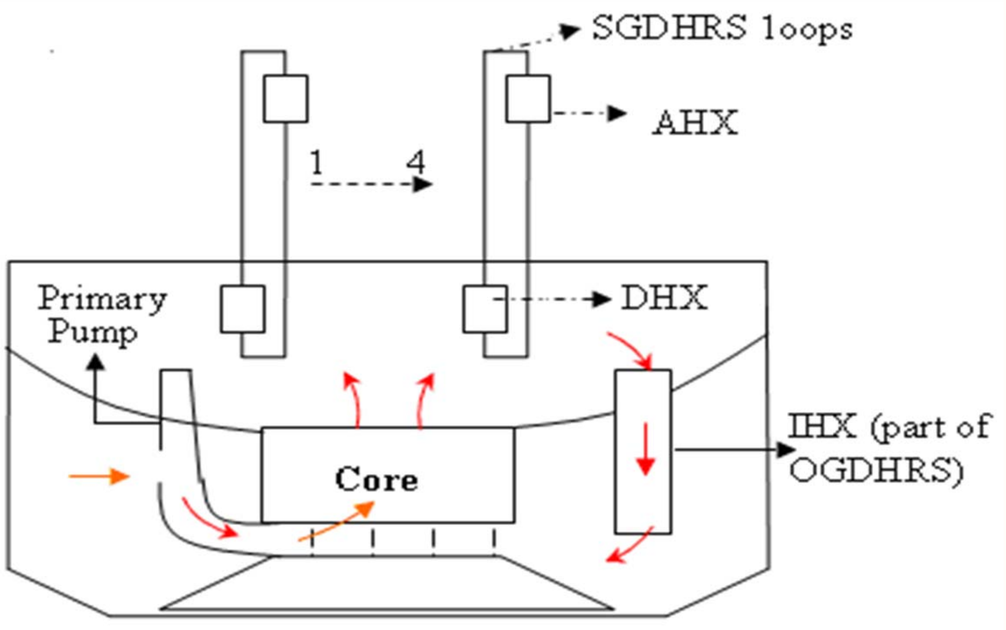
Fig. 1 SHUTDOWN SYSTEM

PFBR Schematic



- Main Heat Transport System
  - Primary Sodium Circuit
  - Secondary Sodium Circuit
  - Spl. DHR heat exchangers with recirculation pump in SWS
- Independent DHR Loops
  - 3% of full power capacity
  - Passive operation
  - Air dampers for startup and freezing prevention

SGDHRS of PFBR



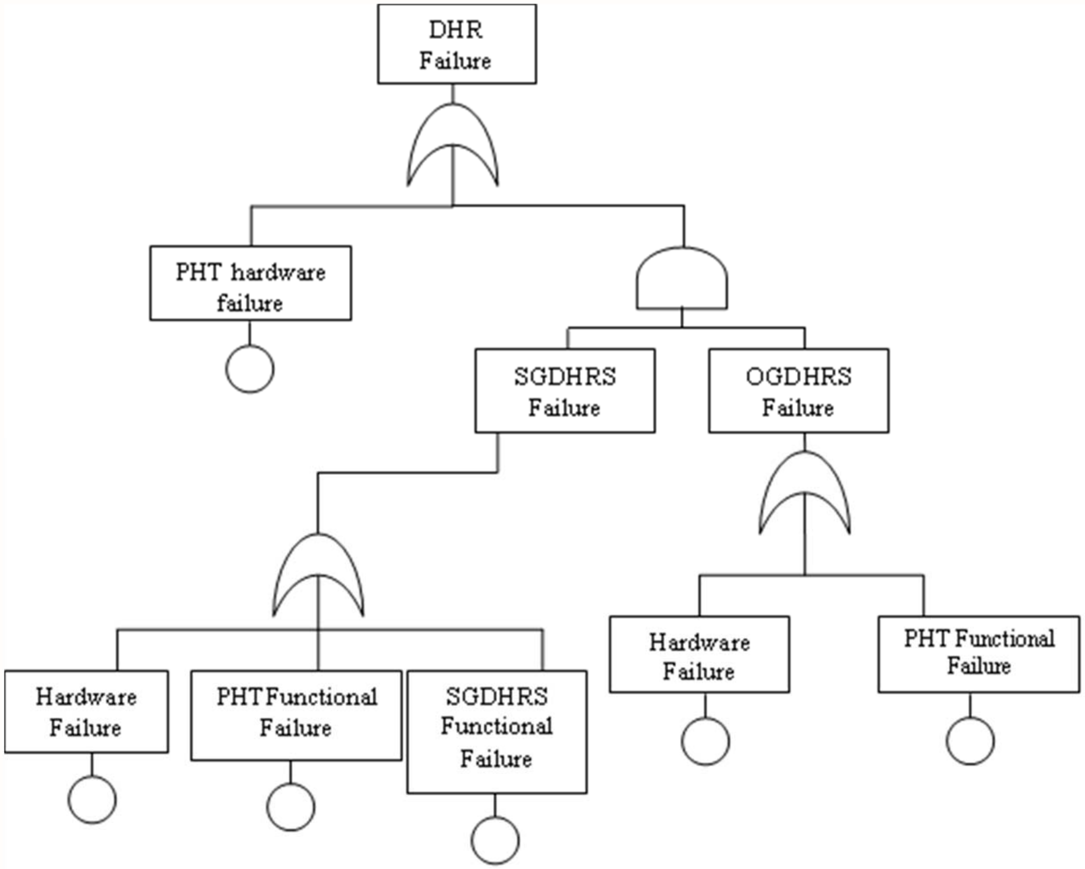
Safety Grade Decay Heat Removal System (SGDHRS) consists of 4 independent thermo-siphon loops, with Na-Na HX dipped in hot pool.

Air dampers in Na-Air Heat exchangers- active components

Primary circuit integrity required for DHR through SGDHRS

Total Failure Probability of DHRS of PFBR

Total DHRS fails if OGDHRS and SGDHRS fails or primary heat transport which is common to both fails



$$\lambda_{DHR-TOT} = \sum_i f_i \begin{pmatrix} P_{PHT,hrd,i} + (P_{OGDHRS,hrd,i} + P_{PHT,fun,OGDHRS,i}) \\ (P_{hrd,SGDHRS,i} + P_{fun,SGDHRS,i} + P_{PHT,fun,SGDHRS,i}) \end{pmatrix}$$

Passive System Reliability-Features/Methods

Thermal Hydraulic Passive System:

- Natural convection driving forces are comparable to resistance

  -Increase in performance uncertainty.

Passive System Reliability/Functional reliability:

- Depends only on functionality or inherent robustness of the process and design margin.
- Since active equipment is absent, reliability of functionality is the dominant factor.
- Various causes like uncertainty in design and critical parameters, design process, environment etc needs to be accounted.
- Quantification methodology is uncertainty propagation and estimation of nominal failure probability. (as in structural reliability studies).

Passive System Reliability - Definition

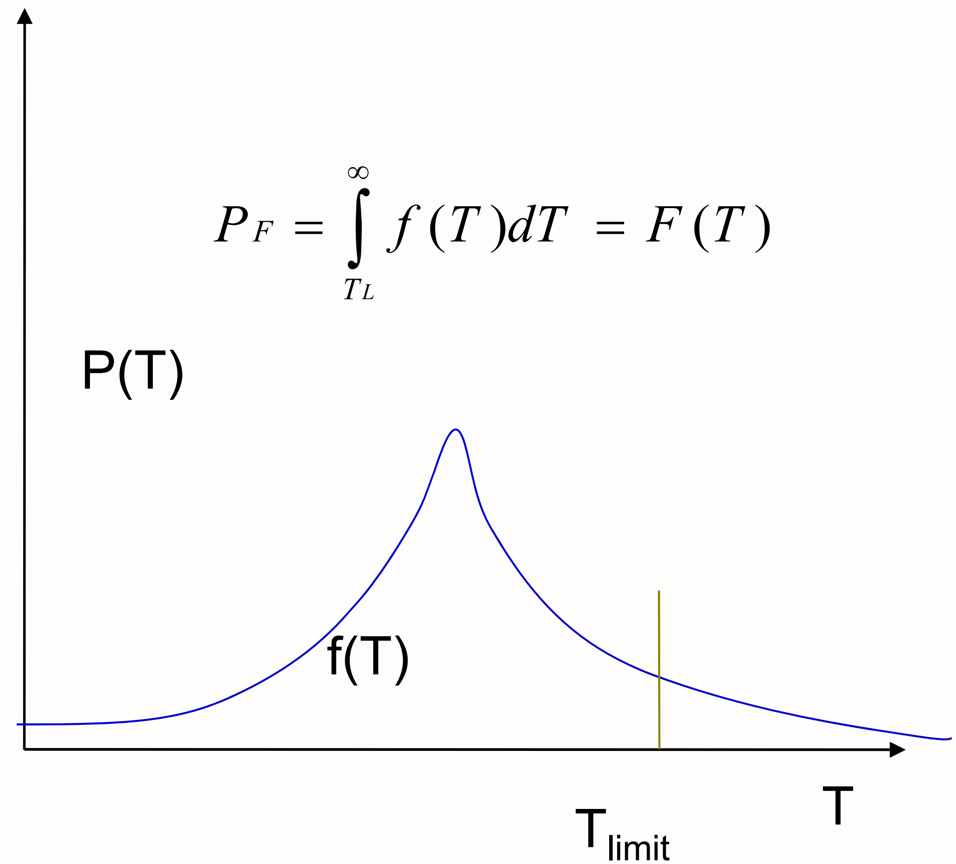
- Focused on Stress-Strength interaction model from structural reliability.

- Reliability $= 1 - P_F$
- (1-Probability of Failure)

$P_F = P\,[T(\text{limit}) - T(x) < 0]$

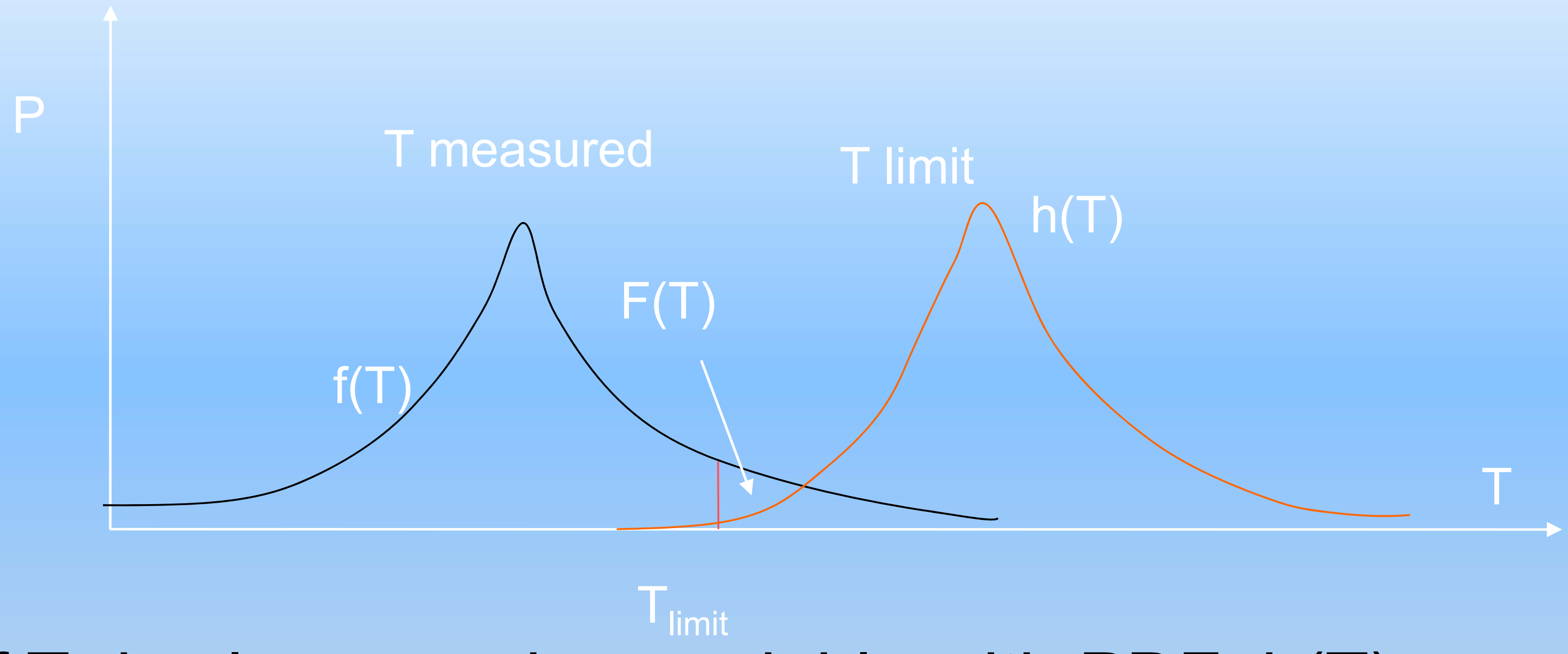
T is a random variable with a pdf f (T)

$$P_F = \int_{T_L}^{\infty} f(T)\,dT = F(T)$$



*A **limit state** function can also be defined for this purpose as L = L(T_L, T)*
*L(T_L, P(**X**))* , since T is calculated as a function of input data X

Passive System Reliability- Definition



If T_L is also a random variable with PDF, h(T)

$$P_F = \int_{-\infty}^{\infty} [h(T)dT \;\; F(T)]$$

Elements of Passive System Reliability Analysis

$P_F =$
$P\,[T(\text{limit}) - T(x) < 0]$

Input parameter pdf

Differential equations

Numerical procedure

Limit state function

$L(x,\alpha) = T(\text{limit}) - T(x, \alpha)$

Steps in Functional Reliability

- System Analysis
- Formulate failure criteria
- Identify significant parameters
  - sensitivity analysis - this step requires almost the end result
- Assign *pdf* to parameters
- Propagate uncertainty
  - large number of calculations - how to do it faster ?
- Calculate exceedence probability/reliability

Uncertainty

- Input parameter uncertainty may have,
  - Aleatoric uncertainty – inherent random variability
  - Epistemic uncertainty – from lack of knowledge
- Can we mix the two?
- Uncertainty is uncertainty
- Aleatoric – pdf is reasonably known from measurement.
- Epistemic – a pdf is judged from expert opinion which could become a delta function
- It could only be a convenient classification

Efficient Propagation of Uncertainty

$$P_F = \int I(x_1, x_2, x_3..x_n) f(x_1, x_2, x_3..x_n) dx$$

- f(x) is *pdf.* I(x) is indicator function,
- I(x) = 1 if the point x_1,..x_n, is a failure point, else 0.
- Computation Time = No. of samples * Time for I(x).
- Direct Monte-Carlo Method: Large fraction of evaluation of I(x) lead to 0.
- Variance Reduction: More samples in region, I(x)=1.
- Approximations to I(x), combined with Variance reduction.

Number Calculations for Reliability Evaluation

- Direct MCS to estimate a Probability of Failure p with fractional error f, the number of calculations
  - $N \sim 1/(p * f^2)$; importance sampling 1/10 to 1/20 reduction;
- To calculate one linear response surface by forward method, number of calculations for n_i parameters

  $N \sim n_i * \text{Time(code)}$
- Adjoint methods require $N \sim n_o * \text{Time(code)}$
- Time (code)= $O(M^2)$, M – problem size
- Adjoint method is useful when $n_o << n_i$
- Combination of efficient approximations with importance sampling appears promising.

Other Strategies for Uncertainty Propagation

- Polynomial chaos methods (N.Wiener)
  - Expansion of input *pdf* in orthonormal basis functions such as Legendre (uniform), Laguerre (exponential), Hermite functions (Gaussian), substitute in DE to get a new set of DE and solve.
- Probabilistic formulation of the differential equations.
- Subset simulations (Au and Beck)
- Line Sampling (Zio et al)

The Role of Adjoint Operators

- Fast approximations to system model to compute limit state functions L = R-S, Combined with variance reduction techniques.
- One such possibility is Taylor series approximation to L(x)

$$L'(x) = L(x_0) + \nabla L \, . \, dx$$

- The components of ∇L(x), can be obtained efficiently by Adjoint methods as follows.

Adjoint Operators – the Method & Gain

Definition: Formal Adjoint

In a linear vector space with inner product,

L* is an adjoint of **L**, If for all vectors **u, v**

$$\mathbf{v}^T \mathbf{L}\, \mathbf{u} = (\mathbf{L}^* \mathbf{v})^T \mathbf{u}$$

Suppose vector '**u**' satisfies $\mathbf{L\, u = f}$; for given **L** and **f** .
We want to evaluate in general some function of **u**, that is the quantity $\mathbf{g}^T \mathbf{u}$.

Find '**v**' such that, $\mathbf{L}^* \mathbf{v} = \mathbf{g}$.
Then from adjoint definition, $\mathbf{g}^T \mathbf{u} = \mathbf{v}^T \mathbf{L u} = \mathbf{v}^T \mathbf{f}$.

That is

$$\boxed{\mathbf{g}^T \mathbf{u} = \mathbf{v}^T \mathbf{f}}$$

The Advantage

The equivalence

$$\mathbf{g}^T \mathbf{u} = \mathbf{v}^T \mathbf{f}$$

has the following implication.

If **Lu = f**, has to be solved for many different 'f' with fixed **g**, then,

The adjoint problem **L* v = g** is solved once and the inner product $\mathbf{v}^T \mathbf{f}$ could be calculated without solving Lu = f, repeatedly.

The advantage

- The number of operations required to repeatedly solve, (L^{-1} is known),
- $Lu = f$ is $O(M^2)$

- With adjoint method only vector product $v^T f$ is required, which is $O(M)$

Adjoint of Simple Differential Operators

Consider the following linear first order ODE,

$$\frac{dn}{dt} = f(t)$$ ....(1) The adjoint **L*** of a differential operator L is defined by

$$\int_{t_1}^{t_2} \varphi(t)\ [L\ \ n(t)]\ \ dt = \int_{t_1}^{t_2} [L^* \varphi(t)]\ \ n(t)dt$$

Multiply equation (1) by φ and integrate by parts

$$\int_{t_1}^{t_2} \varphi \frac{dn}{dt} dt = \varphi(t_2)n(t_2) - \varphi(t_1)n(t_1) - \int_{t_1}^{t_2} n \frac{d\varphi}{dt} dt$$

Therefore formal adjoint operator for $\frac{d}{dt}$ is $-\frac{d}{dt}$

Inorder to satisfy adjoint definition, we require that the boundary terms satisfy the following equation.

$$\varphi(t_2)n(t_2) - \varphi(t_1)n(t_1) = 0$$

Adjoint of simple differential operators

Let the initial condition for solving (1) be, n(0) = α . If f(t) is constant f , solution is

$$n(t) = \int f(t)dt + c$$

$$n(t) = f \quad t + \alpha$$

The solution of the adjoint equation is

$$-\frac{d\varphi}{dt} = g(t) \qquad \varphi(t) = -g \quad t + c$$

$$(-g \quad t_2 + c)(f \quad t_2 + \alpha) = c\,\alpha \qquad g\ t_2\alpha - f\ t_2 c + gf\ t_2^{\,2} = 0 \qquad c = \frac{g}{f}\alpha + gt_2$$

$$\varphi(t) = g \quad (t_2 - t) + \frac{g}{f}\alpha$$

It can be easily verified that a weighted integral of direct function is equal to the integral of '*f*' times the adjoint function.

($g^T u = v^T f$)

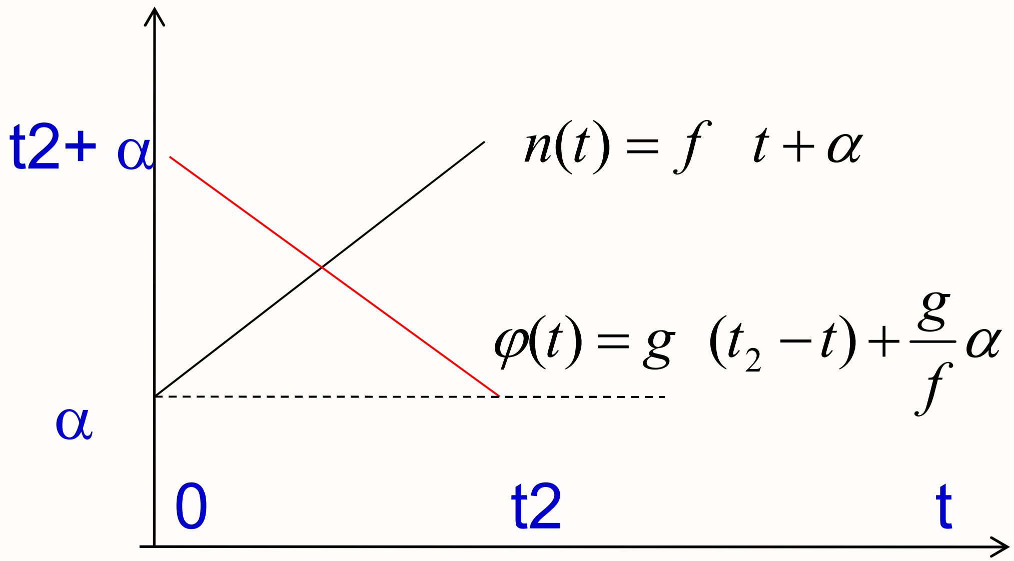
$$\int_0^t n(t')g \quad dt' = \int_0^t \varphi(t')f \quad dt' \qquad = g \quad \alpha \quad t + gf\frac{t^2}{2}$$

Physical interpretation

f=g=1

$\alpha = 1$

t=0..t2



Adjoint calculation in this case is time reversal.

Few physical applications:

particle diffusion:

Forward : given a particle source find particle distribution .

Adjoint : given that a detector is located at some point find source importance distribution.

Structures:

Forward: Load specified, displacement field is calculated.

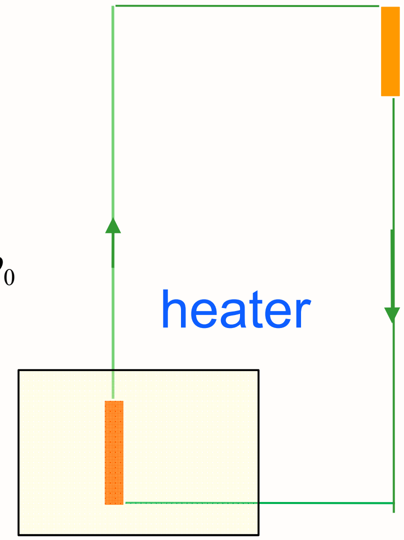
Adjoint : For displacement at specified location find the load

Adjoint Operator – Application to 1-D loop

The equations for fluid mass flow rate 'W'

$$\frac{dW}{dt} = \frac{gA}{L}\oint \rho(T)\cos\theta\, dz - f(W)\frac{W^2}{2\rho D^2}$$

$$\frac{dW}{dt} + B.W^{2-b} = C\int T\cos\theta dx \qquad B = \frac{a\mu^b A^{b-1}}{2D^{1+b}\rho_0} \qquad C = \frac{-A}{L}g\beta\rho_0$$



Forward sensitivity equations:

$$\frac{d}{d\alpha}\frac{dW}{dt} + \frac{dB}{d\alpha}W^{2-b} + B.\frac{dW^{(2-b)}}{d\alpha} = \frac{dC}{d\alpha}\int T\cos\theta dx + C\int\frac{dT}{d\alpha}\cos\theta dx$$

$$\frac{dS_w}{dt} + B(2-b)W^{2-b-1}S_w - C\int S_T\cos\theta dx = \frac{dC}{d\alpha}\int T\cos\theta dx + \frac{dB}{d\alpha}W^{2-b}$$

This eq. has the form $LS_w = f_\alpha$

Adjoint Flow Equations

$$\frac{dS_w}{dt} + B(2-b)W^{2-b-1}S_w - C\int S_T \cos\theta dx = \frac{dC}{d\alpha}\int T\cos\theta dx + \frac{dB}{d\alpha}W^{2-b}$$

Multiplying forward sensitivity equations for the mass flow rate by Z_W (and temperature sensitivity equation by Z_T) and integrating we get,

$$\int_0^\tau Z_w \frac{dS_w}{dt}dt + \int_0^\tau Z_w B(2-b)W^{(1-b)}S_w dt - \int_0^\tau Z_w E\int_0^L S_T \cos\theta \; dxdt = \int Z_w F_w dt$$

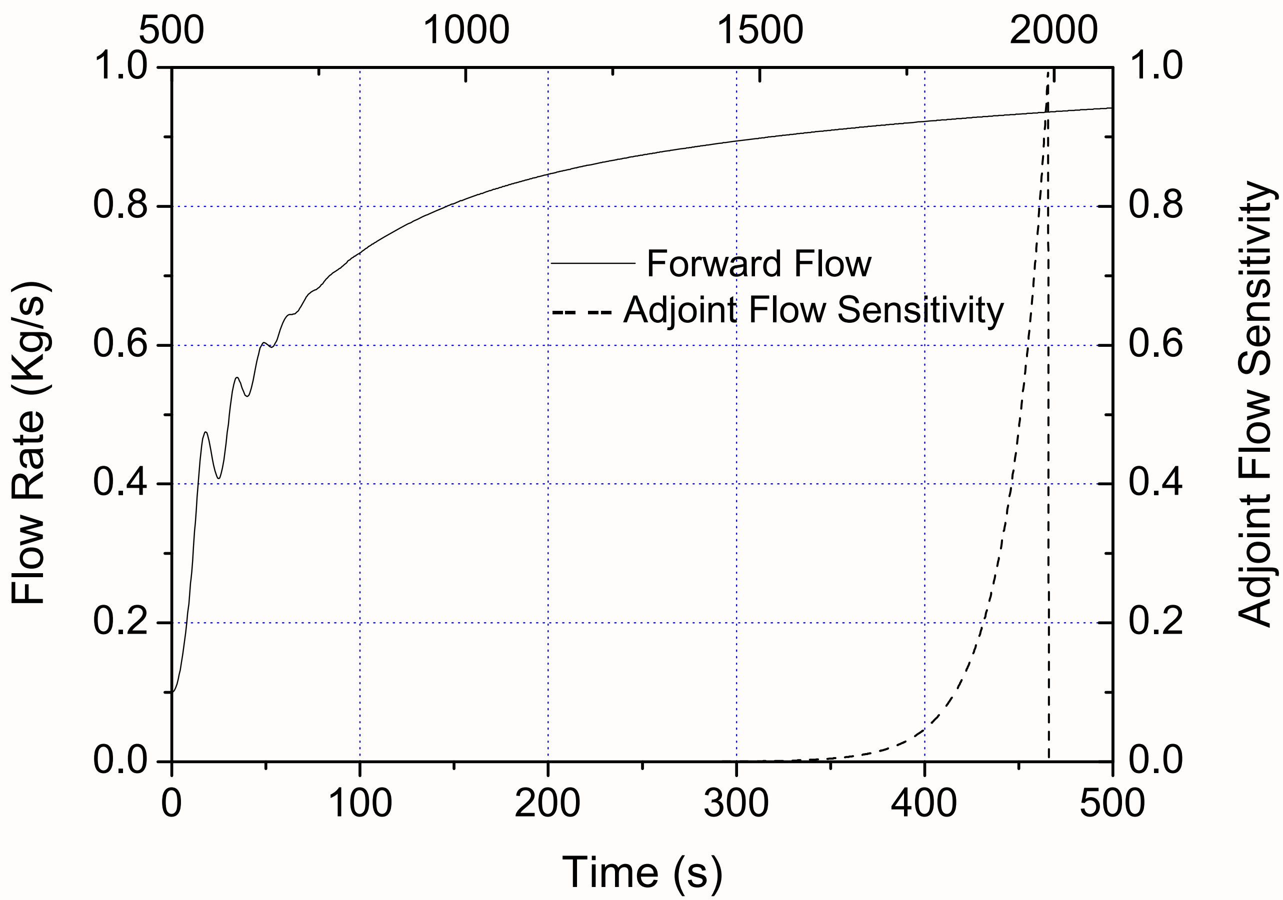
F_W represents the source terms in the forward sensitivity equation. The response function gT u for flow sensitivity is chosen as, $\int g_w(t)S_w(t)dt$

Since $\int_0^\tau S_W g_W dt = \int_0^\tau Z_W F_W dt$

Integrating by parts and rearranging in the following form,

$$-\frac{dZ_W}{dt} + B(2-b)W^{(1-b)}Z_W + \frac{1}{A\rho_0}\int_0^L \frac{\partial T}{\partial x} Z_T dx = g_W(t)$$

Adjoint Function Flow Sensitivity Zw



Sensitivity Results ($\Delta P/\Delta\alpha$) – Linear Approx

If sensitivity is required at a particular point x′ in the loop at time t′, then we require

$$S_W(t') = \frac{dW(t')}{d\alpha}$$

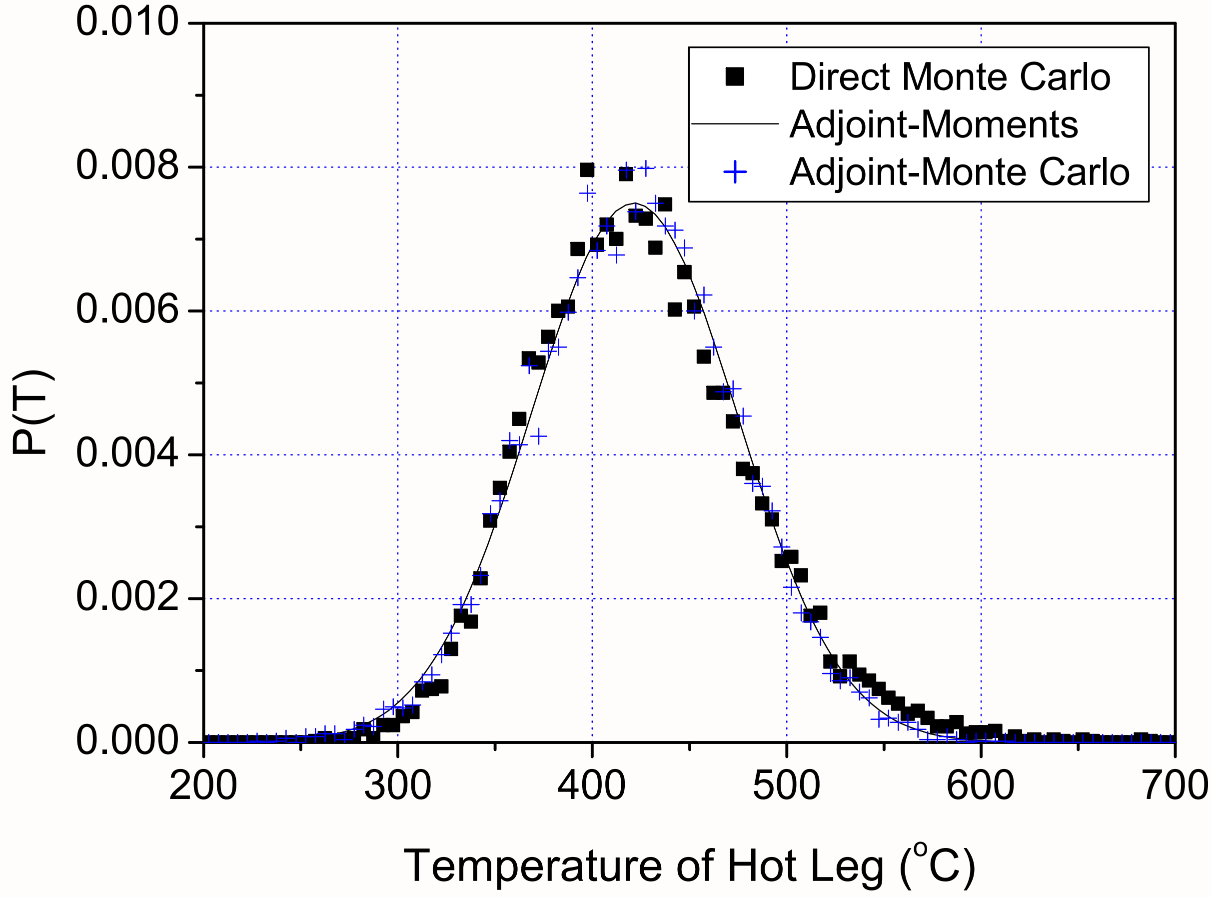
$$S_T(x',t') = \frac{dT(x',t')}{d\alpha}$$

Parameter	Sensitivity on W		Sensitivity on T		Std Dev.
	Finite Difference	Adjoint	Recalculation	Adjoint	
β	*(2128) 2103.7	2129.7	-8309.9	-8661.1	0.05
a	-0.00753	-0.00782	0.031242	0.031765	0.05
μ	-1506.9	-1563.2	6248.4	6352.9	0.05
D	51.127	50.093	-9353.2	-9799.7	0.01
ρ	0.00113	0.001135	-0.01375	-0.01294	0.05
q	4.94E-05	5.00E-05	0.037886	0.037887	0.1
T_0	-4.40E-06	-4.4E-06	0.99455	0.99455	0.1
U	5.66E-06	6.46E-06	-0.35265	-0.36922	0.1

Finite difference: $\Delta P/\Delta X \approx P(X+\Delta X)-P(X) / \Delta X$,

* by direct differentiation of steady state expression. (l=7 m, D= 0.04 m)
Mean temperature of the hot leg = 421.5 ^{o}C. Standard deviation = 53.2 ^{o}C

Probability density function of hot leg temperature at a given time by three different methods



Adjoint Operator Methods -variants

- Differential Eq → Finite Difference → Numerical program

- *Continuous Approach- Discretization of the Adjoint equation.*

- Fully Discrete Method – Adjoint of the Discretized equation.

- *Automatic differentiation – transformation implemented on the program code.*

Automatic Differentiation (AD) Basics

- Program P, Computing Y=P(X)
- Sensitivity by divided difference: $\Delta P/\Delta X \approx P(X+\Delta X)-P(X) / \Delta X$,
  - Truncation Error $O(\Delta x)$.
  - N+1 program runs required if X is a vector of N parameters.

- Automatic Differentiation: Forward (and reverse) derivatives of a program by source code transformation.
- Forward AD: Each line of program code replaced by differentiated statement – forward differentiated code
- Reverse AD: Starting from the End statement, corresponding to each derivative statement, adjoint statement is added to code.
- Reverse program yields derivatives of a response parameter, **for any number** of input parameters in **one run.**
- **T (adj code) > T (forward code), but < N T(forward code)** .
- Difficulty: Required to store program state data of the forward run

Adjoint Code

If P, at some time executes the instruction on variables a, b, c, and array T:

a = b*T(k) + c   statement

Differentiated program executes additional operations on **differentials**

da, db, dc, and array dT(k),

da = db*T(k) + b*dT(k) + dc   differentiated statement

This can be written in matrix form as,

$$\begin{bmatrix} da \\ db \\ dc \\ dT \end{bmatrix} = \begin{bmatrix} 0 & T(k) & 1 & b \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} da \\ db \\ dc \\ dT \end{bmatrix}$$

Adjoint Code

The adjoint of ,

$$\begin{bmatrix} da \\ db \\ dc \\ dT \end{bmatrix} = \begin{bmatrix} 0 & T(k) & 1 & b \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} da \\ db \\ dc \\ dT \end{bmatrix}$$

is

$$\begin{bmatrix} ada \\ adb \\ adc \\ adT \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ T(k) & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ b & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} ada \\ adb \\ adc \\ adT \end{bmatrix}$$

a = b*T(k) + c ← statement

da = db*T(k) + b*dT(k) + dc ← differentiated statement

Adjoint statements are written as,

adb = adb + T(k)* ada ← adjoint statements

adT(k) = adT(k) + b* ada

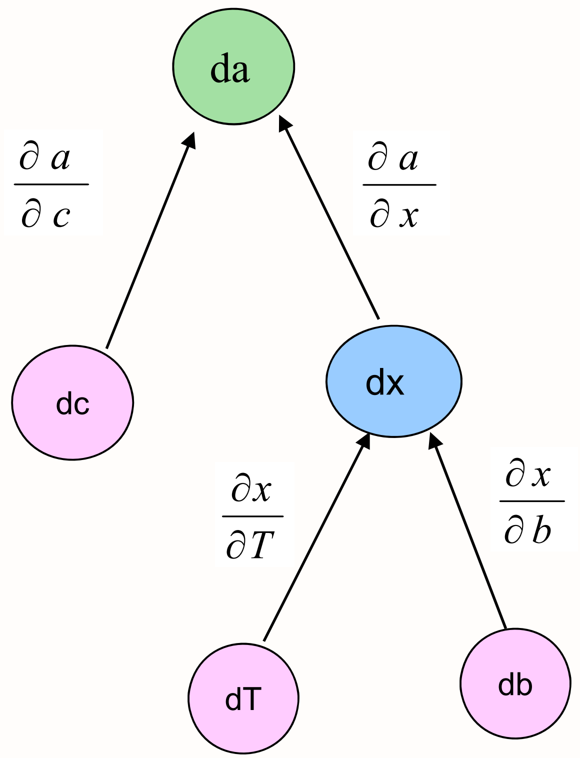
adc = adc+ ada

ada = 0

Flow Graph-Forward Differentiation

Let,
x=b*T(k)
a = x+ c   statements

dx = db*T(k)+b*dT(k)
da = dx + dc   diff statements

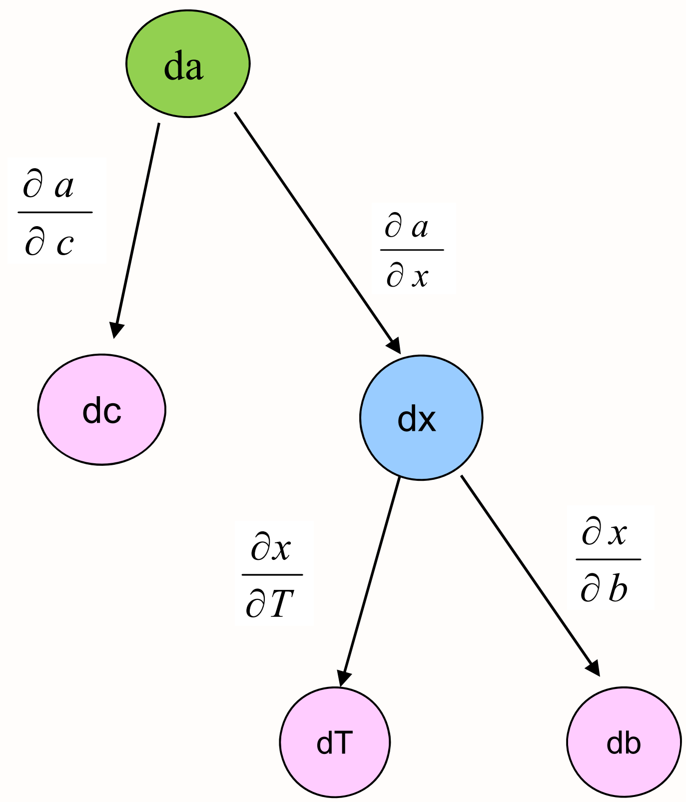


da/dT; dc=0,dT=1,db=0

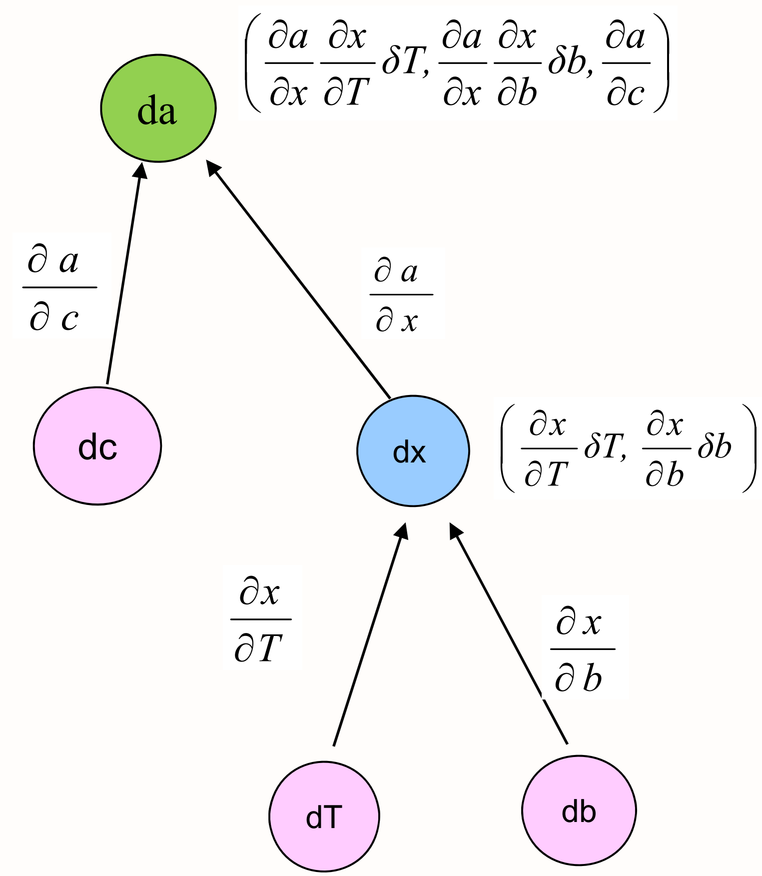
da/db; dc=0,dT=0,db=1

da/dc; dc=1,dT=0,db=0

Flow Graph for Adjoint Differentiation



Reverse flow

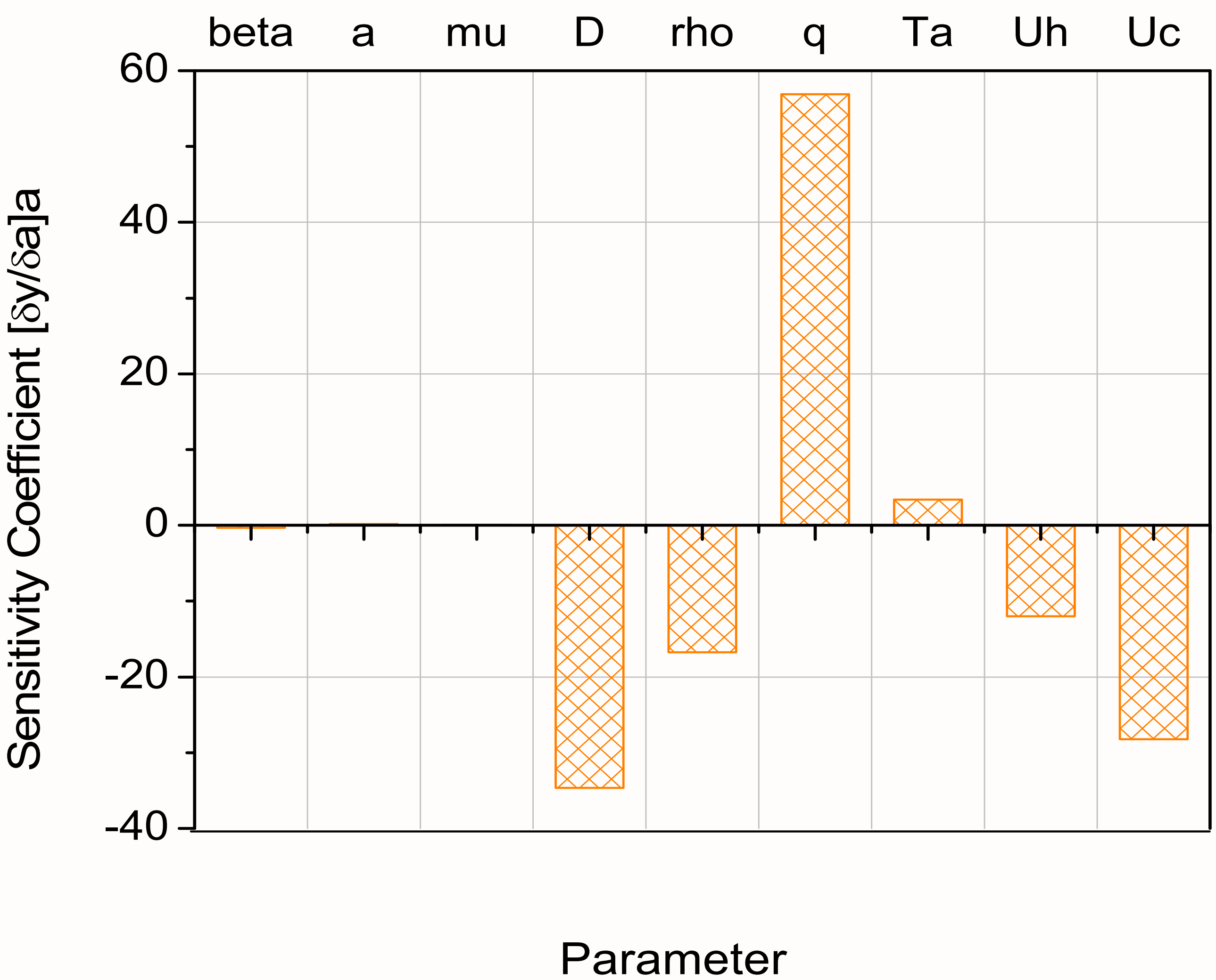


Accumulated/reversible flow

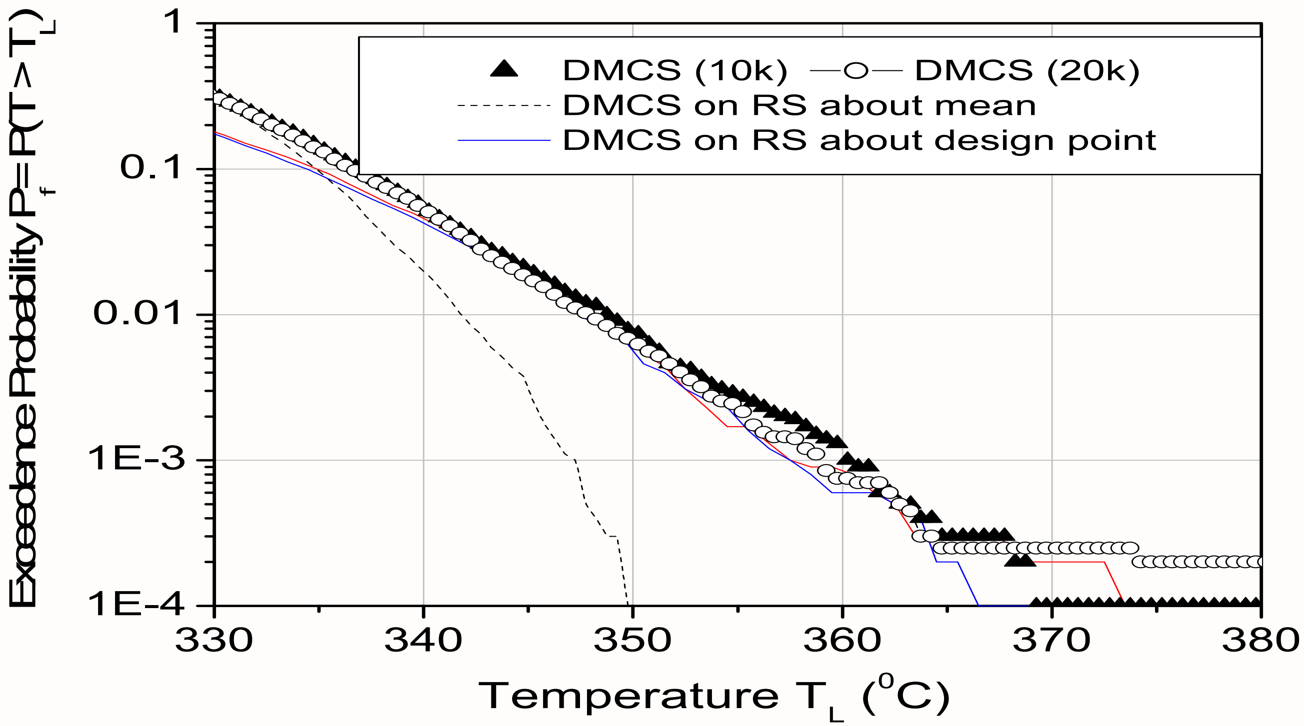
Results from Differentiated Code

Parameter	Sensitivity Coefficients (dT/dα)			
	Finite Difference	Forward Differentiation	Reverse Differentiation	
			Non-Normalised	Normalised
β	-1350.57	-1305.23	-1324.4	-0.31124
a	0.19531	0.18248	0.18331	0.18331
μ	162.125	142.985	143.21	0.04583
D	-690.613	-687.656	-692.94	-34.647
ρ	-0.01886	-0.01868	-0.01902	-16.735
Q	0.00114	0.00114	0.00114	56.876
Ta	0.11241	0.11294	0.11294	3.3881
Uh	-0.05631	-0.05621	-0.00172	-12.007
Uc	--	--	-0.05637	-28.185
Time (s) Intel 1.8 GHz Core 2 Duo	49.5 s	8 x 5.3 s	9.47 s	-

Normalized Sensitivity Coefficients



Probability Distribution for Temp to Exceed 350° C – Different Methods



Probability distribution function for the failure criteria of 350 °C, with operating point response surface MCS, design point response surface MCS and DMCS.

Tools and Methods for Automatic Differentiation

- Existing Tools
  - TAPENADE
  - OpenAD
  - ADIC/ADIFOR

- AD Tool development
  - Lex /Flex (lexical scanner generator)
  - Yacc/Bison (parser generator)
  - Gentle (compiler generator)
  - Development work on a AD tool is in progress*

Conclusion

Demonstration of adjoint operator methods (AD) for,

- Very fast sensitivity analysis and parameter ranking, in time independent of no. of input parameters.
- Speeding up of Passive TH system reliability analysis.
- Efficient Best Estimate Plus Uncertainty Analysis
- Alternate Importance sampling MCS (utilizing approx prediction of failure region) promising for complex/arbitrary failure regions is being explored.

REFERENCES

- Michael b. Giles and Niles A. Pierce, An Introduction to the Adjoint Approach to Design, *Flow, Turbulence and Combustion* ***65: 393–415, 2000,*** *Kluwer Academic Publishers.*

- Yang Cao, Shengtai li, Linda petzold, and Radu Serban, Adjoint Sensitivity Analysis for Differential-algebraic equations: the adjoint DAE system and its numerical solution, Siam j. Sci. Comput., Vol. 24, no. 3, pp. 1076–1089, 2003.

- Compiler Construction Principles and Practice, Kenneth C. Louden, CENGAGE Indian reprint, 2008.
- R. Giering and T. Kaminski, Receipes for Adjoint Code construction. Max-Plank institute for meteorology.

Thank you

END OF PRESENTATION

Temperature Sensitivity

$$\frac{\partial T}{\partial t} + \frac{W}{A\rho}\frac{\partial T}{\partial x} = \frac{4q(x)}{D\rho\ C_p} - \frac{4U(x)}{D\rho\ C_p}(T - T_0)$$

$$\frac{\partial S_T}{\partial t} + \frac{1}{A\rho_0}\frac{\partial T}{\partial x}S_w + \frac{w}{A\rho_0}\frac{\partial S_T}{\partial x} + C_2US_T = \frac{\partial}{\partial\alpha}\left(\frac{w}{A\rho_0}\right)\frac{\partial T}{\partial x} + \frac{d}{d\alpha}(C_2U)T + \frac{d}{dx}\left[C_1q(x) + C_3V(x)T_0\right]$$

$$\frac{\partial}{\partial t}\frac{\partial T}{\partial\alpha} + \frac{\partial}{\partial\alpha}\left(\frac{w}{A\rho_0}\right)\frac{\partial T}{\partial x} + \frac{w}{A\rho_0}\frac{\partial}{\partial\alpha}\frac{\partial T}{\partial x} + T\frac{\partial}{\partial x}(C_2U(x)) + C_2U(x)\frac{\partial T}{\partial\alpha} = \frac{d}{d\alpha}\left(C_1q(x) + C_3U(x)T_0\right)$$

The response function gT u for flow sensitivity is chosen as,

$$\int g_w(t)S_w(t)dt$$

In this case, $g_w(t)$ is taken as delta function.

Adjoint Sensitivity Equations

Multiplying forward sensitivity equations for the mass flow rate by Z_W and temperature sensitivity equation by Z_T and integrating we get,

$$\int_0^{\tau} Z_w \frac{dS_w}{dt} dt + \int_0^{\tau} Z_w B(2-b) W^{(1-b)} S_w dt - \int_0^{\tau} Z_w E \int_0^{L} S_T \cos\theta \; dxdt = \int Z_w F_w dt$$

$$\int_0^{L}\int_0^{\tau} Z_T \frac{\partial S_T}{\partial t} dtdx + \int_0^{L}\int_0^{\tau} Z_T \frac{W}{A\rho_0} \frac{\partial S_T}{\partial x} dtdx + \int_0^{L}\int_0^{\tau} Z_T CUS_T dtdx + \int_0^{L}\int_0^{\tau} Z_T \frac{1}{A\rho_0} \frac{\partial T}{\partial x} S_W dtdx = \int_0^{L}\int_0^{\tau} Z_T F_T dtdx$$

With F_T and F_W representing the source terms in the forward sensitivity equations.

$$\int_0^{\tau} S_W g_W dt + \int_0^{L} \int_0^{\tau} S_T g_T dtdx = \int_0^{\tau} Z_W F_W dt + \int_0^{L}\int_0^{\tau} Z_T F_T dtdx$$

Integrating by parts and rearranging in the following form,

Adjoint Sensitivity Equations

The following equations are obtained for g_w and g_T,

$$-\frac{dZ_W}{dt} + B(2-b)W^{(1-b)}Z_W + \frac{1}{A\rho_0}\int_0^L \frac{\partial T}{\partial x} Z_T dx = g_W(t)$$

and

$$-\frac{\partial Z_T}{\partial t} - \frac{W}{A\rho_0}\frac{\partial Z_T}{\partial x} + EUZ_T - C\cos\theta \quad Z_W = g_T(x,t)$$

The identity **gT u = vT f** is,

$$\int_0^\tau S_W g_W dt + \int_0^L \int_0^\tau S_T g_T dt dx = \int_0^\tau Z_W F_W dt + \int_0^L \int_0^\tau Z_T F_T dt dx$$

The sensitivity expressions on LHS are evaluated from Z_W and Z_T in the RHS. This results in computational cost reduction, when the number of sensitivity variables are many compared to the out put response variable.

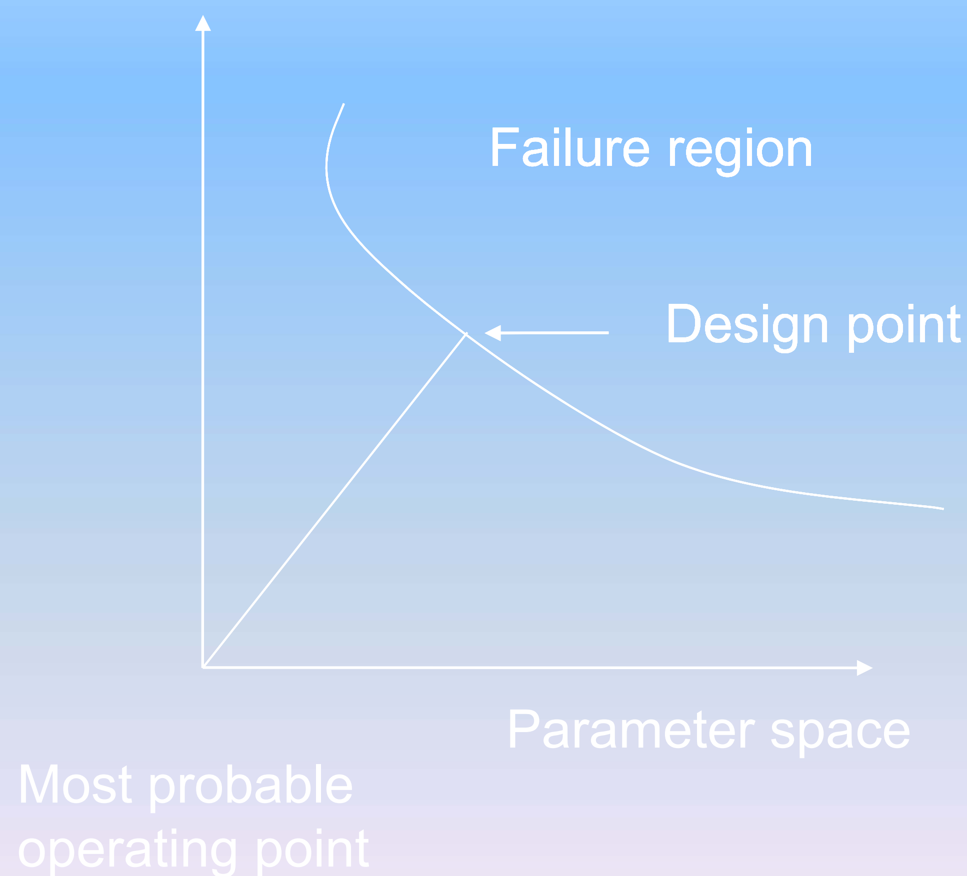
Comparison of Computational Efficiency for Different methods

Method	**Number of simulations**	**$P_f(T > T_L = 340)$**	**$P_f(T > T_L = 350)$**	$\frac{\sigma(P_f)}{E(P_f)}\sqrt{N}$
DMCS	20000	0.056	6.65E-3	12.72
RS-MC-MCS-N	700-1000	-	7.15E-3	4.6
RS-MCS-N	1000	-	7.35E-3	4.7
RS-MCS-HPN	1300	-	6.60E-3*	2.8
RS- MC-MCS-HPN	700,1000	-	6.28E-3,6.62E-3*	3.3-3.12

Speed up ~ ratio of Δ^2

Importance Sampling Markov Chain MCS

- Importance sampling of failure region to reduce variance
- Locate failure region with the help of Linear approximations to the program code iteratively.
- Use Metropolis algorithm for generating samples.



Markov Chain MCS

Markov – Chain Monte Carlo for generating samples

Easy to implement and efficient.

Possibility to utilize gradient information for better convergence.

Uses Markovian property to draw set of samples $\{\boldsymbol{x}_k\}$ from target pdf, $q(\boldsymbol{x})$.

The Metropolis algorithm

Select initial parameter vector $\boldsymbol{x}_0$.

Draw trial step from a symmetric pdf, i.e., $t(\Delta\, \mathbf{x}) = t(-\Delta\, \mathbf{x})$

Iterate as follows: at iteration number k

create trial position $\boldsymbol{x}' = \boldsymbol{x}_k + \Delta\boldsymbol{x}$, $\Delta\boldsymbol{x}$ randomly chosen from $t(\Delta\boldsymbol{x})$

calculate ratio $r = min\ \{ 1, q(\boldsymbol{x}')/q(\boldsymbol{x}_k)\}$ *and for a random number U*

if $(U \leq 1)$, set $\boldsymbol{x}_{k+1} = \boldsymbol{x}'$ else $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k$