國立成功大學
機械工程學系
碩　士　論　文

隨機取樣 SQP 演算法以解決非線性可靠度限制式之最佳化問題

A Filter-Based Sample Average SQP For Optimization Problems with Highly Nonlinear Probabilistic Constraints

研 究 生：許凱勛
指導教授：詹魁元 博士

中 華 民 國 九 十 八 年 七 月

隨機取樣 SQP 演算法以解決非線性可靠度限制式之最佳化問題

A Filter-Based Sample Average SQP For Optimization Problems with Highly Nonlinear Probabilistic Constraints

研 究 生：許凱勛　　Student: Kai-Hsun Hsu

指導教授：詹魁元 博士　　Advisor: Dr. Kuei-Yuan Chan

國立成功大學

機械工程學系

碩士論文

A Thesis

Submitted to Department of Mechanical Engineering

National Cheng Kung University

in Partial Fulfillment of the Requirements

for the Degree of

Master of Science

in

Dept. of Mech. Eng.

June 2009

Tainan, Taiwan

中華民國九十八年七月

隨機取樣 SQP 演算法以解決非線性可靠度限制式之最佳化問題

學生： 許凱勛　　指 導 教 授 ： 詹魁元 博士

國立成功大學機械工程學系

摘　　要

本研究爲解決非線性可靠度限制式之最佳化問題，提出一個以樣本法與 SQP 爲骨架的演算法。此演算法使用平均重要取點法(AAIS)來估計破壞機率與其梯度。平均重要取點法爲取樣在函數的極限狀態上，因此可使用較少的樣本數得到穩健且精確的破壞機率估計值。此外，平均重要取點法也可在有限的樣本數下，得到相對精確且變異量小的梯度量值。爲防止破壞機率梯度的計算結果遭受機率值低但梯度值高的樣本影響，本研究提出一樣本過濾器，防止少數樣本導致梯度計算結果產生偏差。爲確保演算法收斂，本演算法中使用一試驗準則(F-filter)來替代懲罰函數，用以判斷設計點可否被更新，並避免使用懲罰函數所造成的不良條件。 爲了提升演算法的效率，將部分已取樣之樣本回收再利用，做爲下一次疊代的樣本，因此下一次疊代只需在上一次疊代過程中還未取樣的區域進行取樣即可。文中以一個工程範例以及兩個數學範例說明以本研究所提出演算法之計算結果並與文獻上的方法加以比較，其中一個數學範例爲有多個最大可能破壞點，使用現行一階、二階可靠度方法與蒙地卡羅法皆無法有效的解決。本文提出的演算法除了可有效的解決多破壞模式問題，也可提昇處理一般可靠度最佳化問題的效率、穩健度，及收斂性。對於工程上存在的高度非線性可靠度限制式，本研究所提出的取樣 SQP 演算法較文獻上的方法有著更優異的演算性能。

A Filter-Based Sample Average SQP For Optimization Problems with Highly Nonlinear Probabilistic Constraints

Student: Kai-Hsun Hsu Advisor: Dr. Kuei-Yuan Chan

Department of Mechanical Engineering
National Cheng Kung University

ABSTRACT

In this work we extend a filter-based sequential quadratic programming (SQP) algorithm to solve reliability-based design optimization (RBDO) problems with highly nonlinear constraints. This filter-based SQP uses the approach of average importance sampling (AAIS) in calculating the values and the gradients of probabilistic constraints. AAIS allocates samples at the limit state boundaries such that relatively few samples are required in calculating constraint probability values to achieve high accuracy and low variance. The accuracy of probabilistic constraint gradients using AAIS is improved by a sample filter to eliminate sample outliers that have low probability of occurrence and high gradient values. To ensure convergence, this algorithm replaces the penalty function by an iteration filter to avoid the ill-conditioning problems of the penalty parameters in the acceptance of a design update. A sample-reuse mechanism is introduced to improve the efficiency of the algorithm by avoiding redundant samples. 'Unsampled' region, the region that is not covered by previous samples, is identified by the iteration step lengths, the trust region, and constraint reliability levels. As a result, this filter-based sampling SQP can efficiently handle highly nonlinear probabilistic constraints with multiple most probable points or functions without analytical forms. Several examples are demonstrated and compared with FORM/SORM and Monte Carlo simulation. Results show that by integrating the modified AAIS with the filter-based SQP, overall computation cost can be significantly improved in solving RBDO problems.

誌　　謝

首先我要感謝我的指導教授 詹魁元教授，在兩年研究所學習的過程當中給予指導與鼓勵，讓學生受益匪淺不論在課業或是待人接物上都有了啓發，對於老師的教誨，學生銘記於心，感謝口試委員：陳家豪教授與廖國偉教授對本論文的指導與建議。

在就讀碩士的兩年當中，有許多好朋友好夥伴陪著我渡過，不論好與壞的時光。感謝已畢業的學長：岱璟、智豪在碩一時給予的關心與照顧，也謝謝實驗室的夥伴們：東信、彥智、琇雯．昱達、子頡、勝昌、佳豪、季儒、祐伸與 CMD 實驗室的大家，謝謝你們一路的支持與陪伴，總是帶給我許許多多的歡樂。另外我也很感謝 明倫、宜華、思賢，你們總在我需要幫助的時候，從不吝惜的幫助我，沒有你們我可能無法走的這麼遠。

最後我要謝謝我的父母親與姊姊，對我無怨無悔的付出，沒有你們，就沒有現在的我，本論文僅獻給我最親愛的你們。

目　　錄

表　目　錄

圖　目　錄

符 號 說 明

$\mathbf{d}$	設計變數
$\mathbf{d}^*$	最佳解
F_X	隨機變數 X 的累積分佈函數(CDF)
f_X	隨機變數 X 的機率密度函數(PDF)
$f^* = f(\mathbf{d}^*, \mathbf{p})$	目標函數的最佳值
$g_j(\mathbf{d}, \mathbf{X})$	非等式的拘束條件
$I_g(\mathbf{d}, \mathbf{X})$	指標函數
N	隨機取樣的樣本數
P_{f}	破壞機率
$P_{\mathrm{f},j}$	拘束條件 g_j 的破壞機率
$\hat{P}_{\mathrm{f}}$	破壞機率的估計值
$\hat{P}_{\mathrm{f},j}$	拘束條件 g_j 破壞機率的估計值
$P_{\mathrm{f}}^{\mathrm{allow}}$	最大可容許的破壞機率
$\mathrm{Pr}[\cdot]$	$[\cdot]$的機率
Re	可靠度
$\hat{\mathrm{R}}\mathrm{e}$	可靠度估計值
S_{F}	安全係數
$T(\mathbf{d}, \mathbf{r})$	懲罰函數
t	t%樣本過濾器
U	標準化的高斯隨機變數

$\bar{\mathbf{U}}$	除了控制變數以外的隨機變數
$\bar{\mathbf{u}}$	從隨機變數$\bar{\mathbf{U}}$ 中取的樣本
$\mathrm{Var}[\cdot]$	$[\cdot]$的變異量
v	拘束條件中最大的違反量
X	隨機變數 X
x_i	從母體 X 中取的樣本
α	試驗法則內的參數
β	可靠度指標
$\mathbf{\Delta}$	均勻分佈的範圍參數
Δf	眞實的目標函數減少量
Δq	估計的目標函數減少量
ϵ	放寬參數的標準差
γ	試驗法則內的參數
κ	曲率
μ_X	隨機變數 X 的平均值
$\nabla_{\mathbf{d}} g_j$	拘束條件 g_j 對設計變數的梯度
Ω^k	第 k 次疊代取樣的範圍
Ω^{k+1}	第 $k+1$ 次疊代回收樣本的範圍
Φ	標準高斯的累積分佈函數
ϕ	標準高斯的機率密度函數
ρ	信賴區間(Trust Region)
σ_X	隨機變數 X 的標準差
ζ	試驗法則內的參數

第 一 章　研究背景及動機

設計是一連串策略決定的過程 [1]。在產品設計的初期，設計工程師必須制定許多產品或系統的規格，一旦制定，這些規格也深深影響產品的製作成本、使用時期的長短及期間內的性能表現、乃至於回收或再利用的容易程度。根據文獻顯示，70%的製造成本及產品性能取決於設計初級所作的決定 [2]，也因此工程師必須在最多資訊的情況下做最佳的設計決策。

最佳化設計是工程師時常用來設計產品或系統的工具之一，在完成模型的建構及變數選擇後，最佳化可以有系統的找到滿足設計條件的最佳設計。然而這些設計結果乃根據特定模型及特定假設下所得到的結果，一旦產品所處的環境與當初設計的不同，或有其它的不確定因素(uncertainty)存在，該設計結果便不再是最佳。如許多參數如材料性質等，在設計過程常視爲定值，但實際上仍會變動。此外，最佳設計往往將產品性能推展到極致，這樣的設計常在某設計限制的臨界，一旦產品受到人爲或非人爲、可控制或不可控制的不確定因素影響時，其可靠度將相對較低。因此原本的最佳化問題需再考慮不確定因素，這便是可靠度最佳化設計問題(reliability-based optimum design, RBDO)所訴求的宗旨。可靠度最佳化設計正視不確定因素的存在，故讓工程師在設計優良產品的同時，也考量到產品使用時的可靠度表現。可靠度最佳化設計已在許多工程領域都廣泛受到關注，如:土木 [3]、機械 [4]、化工 [5]、航空 [6]都有其蹤影。

1.1　研究背景

可靠度最佳化設計的源起可從最佳化與可靠度兩方面來探討，最佳化含括了設計的意念，而可靠度則是評估產品或系統表現的一種指標，因此可藉由可靠度的評估後，利用最佳化來改善該產品或系統的表現，這樣的持續、反覆的操作就是處理可靠度最佳化設計問題的過程。因此以下將針對“最佳化”與“可靠度”兩方面來說明可靠度最佳化設計問題的背景。

1.1.1 最佳化概述

最佳化簡單來說就是滿足某些特定的限制條件下，所能得到最好的目標結果。以一個結構設計問題爲例，一般工程師在設計橋樑時，除了達到交通運輸目的之外，還會考量橋樑結構本身強度是否足夠承受一些外力負載，使橋樑不致破壞或是崩塌，並且期許用最少的成本來完成這項工程建設。成本就是工程師的目標，須能承受某些外力負載就是其限制。相同的工程問題可能因爲時空、環境或是目標的不同，而有不同的最佳化問題面貌。舉例來說，若工程師的目標是要打造一座可以使用最久的橋，此最佳化的目標便由成本改爲使用年限。一旦最佳化形式改變，最佳結果也隨之不同。在某些狀況下，設計者需同時考量兩個以上的目標，這便是文獻上的多目標最佳化問題 [7]。本論文不針對多個目標的最佳化問題來做討論，只討論單一目標函數的最佳化問題。方程式(1.1)爲最佳化問題的一般數學模式，其中包含目標函數 $f(\mathbf{d})$ ，等式拘束限制式 $\mathbf{h}(\mathbf{d})$，和非等式限制式 $\mathbf{g}(\mathbf{d})$。

$$\begin{aligned} \min_{\mathbf{d}} \quad & f(\mathbf{d},\mathbf{p}) \\ \text{subject to} \quad & \mathbf{g}(\mathbf{d},\mathbf{p}) \leq 0 \\ & \mathbf{h}(\mathbf{d},\mathbf{p}) = 0 \end{aligned} \tag{1.1}$$

目標函數在此希冀其越小越好，若目標是希望越大越好時，只需將 f 加負號即可。拘束條件 $\mathbf{g}$ 和 $\mathbf{h}$ 均爲需滿足的設計條件，方程式(1.1)使用全負型態(negative null form)，即 $\mathbf{g} \leq 0$ 爲滿足拘束條件的安全設計，而 $\mathbf{g} > 0$ 稱爲違反(violation)亦或是破壞(failure)。在設計過程中會隨疊代過程而更迭的設計量值爲設計變數 $\mathbf{d}$(design variable)，不變的稱爲設計參數 $\mathbf{p}$(design parameter)， 滿足所有拘束條件的解集合表示爲 $\mathcal{F}$ 稱爲可行解空間(feasible space)。系統化的尋找最佳解的疊代機制爲演算法，而每次疊代過程的設計變數稱爲設計點，經由演算法最終獲得的設計變數結果爲最佳解($\mathbf{d}^*$)(optimum)，最終的目標函數值爲最佳值($f^* = f(\mathbf{d}^*, \mathbf{p})$)。

$$\min T(\mathbf{d},\mathbf{r}) = f(\mathbf{d}) + F(\mathbf{g}(\mathbf{d}), \mathbf{h}(\mathbf{d}), \mathbf{r}) \tag{1.2}$$

在演算的過程中，有時因提昇效率而允許設計點違反拘束條件以換取目標函數的提昇，文獻上常用的解決方法爲將目標函數值及違反的拘束條件量值形成一如方程式(1.2)的懲罰函數(penalty function) [1]。其中 F 爲懲罰項而 $\mathbf{r}$ 爲該懲罰項的控制量。許多文獻採用逐漸放大懲罰項的概念將設計點循序導入可行解空間，然而此一作法會使整體最佳化目標 T 具有數值上的不良條件(ill-condition)，亦即當演算法逐漸收斂，T 也變得越來越難尋找最佳解。

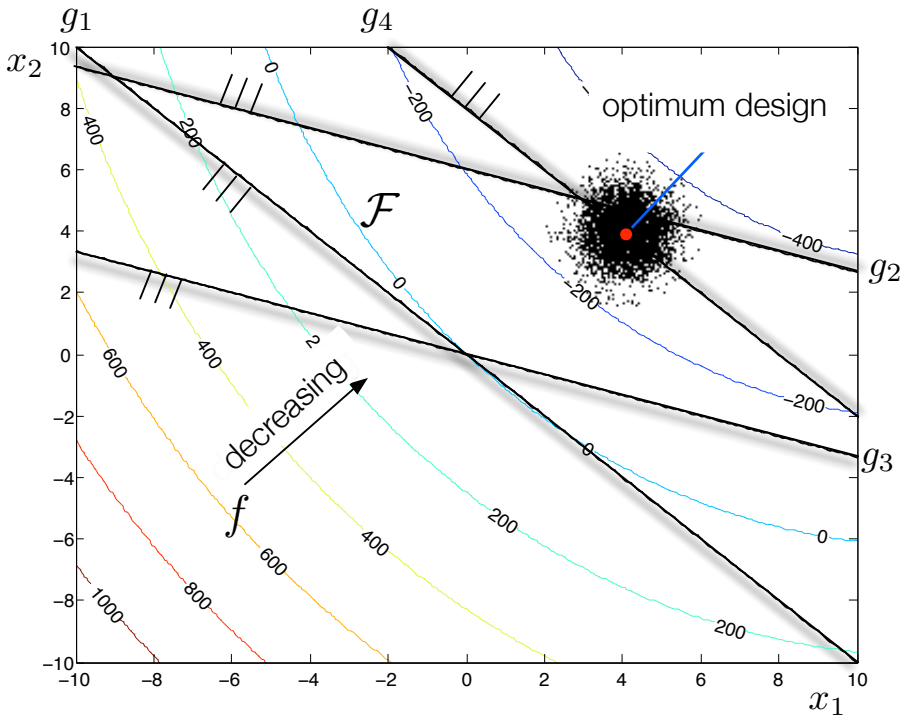

圖 1.1: 不確定因素影響最佳化設計

如何加入“適當”的懲罰項在文獻上有被廣泛的討論且是一個棘手的難題 [1]。

除了不良條件形成的演算困難之外，方程式(1.1)所設計出的產品也有工程上可靠度的問題。爲了讓目標函數達到極致，方程式(1.1)的最佳解往往在拘束條件的臨界。以圖1.1的二維設計問題爲例，由四個拘束條件 g_1 至 g_4 組成一可行解空間 $\mathcal{F}$，而最佳解座落在拘束條件 g_4 上並相當靠近拘束條件 g_2，若有不確定因素，則最佳解對拘束條件 g_4 與 g_2 的可靠度分別爲 50% 與 70%，由此可知最佳解受不確定素影響，造成違反拘束條件的可能性，見文獻 [3–5]。因此，由圖1.1可知，即使最終的最佳化問題不違反拘束條件，也不代表在眞實世界中，經最佳化過程得到的結果，一定能保証該拘束條件一定不會被違反，這句話的意函利用之前的結構問題來做說明。在設計橋樑的最佳化問題中，若拘束條件訂定爲結構強度大於外力負荷，經最佳化設計的橋樑就永遠不會毀壞了嗎?很明顯的答案是否定的，因爲永不毀壞的橋並不存在，那是什麼原因造成這樣的矛盾呢？就是“不確定因素”，也就是因爲有不確定因素的存在，才需評估可靠度。

1.1.2 可靠度概述

可靠度是產品在指定時間範圍內可滿足某性能指標的機率 [5]，也是評估不確定因素影響產品或系統的重要指標。不確定因素在工程上扮演一個很重要的角色，例如當工程師設計橋樑時，可能需考慮地震、颱風的發生性，或材料本身的不確定性，所造成橋樑破壞的機率。因此如何將不確定因素納入考量並解決，成爲一個嚴峻的挑戰。在既往的歷史中，工程師面臨不確因素時，往往使用安全係數來處理這問題，一般可寫成如方程式(1.3)的形式

$$S_{\mathrm{F}} = \frac{R_{\mathrm{N}}}{S_{\mathrm{N}}} \tag{1.3}$$

其中 S_{F} 爲安全係數(safety factor)，若以結構問題爲例，則 R_{N} 爲結構體最大的抗拉強度， S_{N} 爲實際運作或正常負荷之安全容許抗拉強度。若在先前橋樑設計問題時引入安全係數的觀念，則 R_N 爲橋樑抵抗外力的能力，而 S_{N} 爲正常情況時橋樑所承受的負載，因此橋樑問題，可改寫成方程式(1.4)。安全係數的用意就是用比原先高上數倍的標準，來防止不確定因素不會影響產品或系統的表現，但使用比原先高上數倍的標準，往往會造成成本的增加或對產品的其它性能造成影響。在使用安全係數的概念時，需先訂定安全係數 S_{F} ，一般是由工程師所訂定的，因此如何訂定安全係數，往往是依工程師的經驗來訂定的，這樣決定下所設計出的產品便不夠穩健，太倚賴工程師對產品的經驗與評估，若工程師經驗不足或資訊不足時，便容易做出錯誤的決策。因此希望能藉由統計的方法與發生的機率，來輔助工程師了解眞實的情形，工程師便能做出更好的判斷。

$$\begin{aligned} \min \quad & \text{重量或成本} \\ \text{subject to} \quad & S_{\mathrm{F}} S_{\mathrm{N}} - R_{\mathrm{N}} \leq 0 \end{aligned} \tag{1.4}$$

1.1.3 可靠度最佳化設計

可靠度最佳化設計融合了最佳化與可靠度。在最佳化目標函數的要求下考慮不確定因素下的拘束條件，一般用方程式(1.5)表示。本文章爲求符號精簡，將設計參數省略，並再次定義符號，首先是設計變數；令設計變數的集合爲 $\mathbf{d}$ ，設計變數又可分爲定值的設計變數(deterministic design variable) $\mathbf{x_d}$ 及考慮不確定因素的設計變數(random variable) $\mathbf{X}$ 。在此皆以高斯分佈來模擬不確定因素，因此考慮不確定因素的設計變數可表示爲 $\mathbf{X} = \boldsymbol{\mu}_{\mathbf{X}} + \boldsymbol{\sigma}_{\mathbf{U}}$ ，其中 $\boldsymbol{\sigma}_{\mathbf{X}}$ 是標準差，爲常數參數， $\mathbf{U}$ 爲平均值0，標準差爲1的高斯隨機變數。而 $\mathbf{P}_{\mathrm{f}}^{\mathrm{allow}}$

爲最大允許的破壞機率，爲設計者自行給定，值愈小則條件愈嚴苛，反之亦然。P_{f} 爲眞實拘束條件的破壞機率，而可靠度爲 $\mathrm{Re}=1-P_{\mathrm{f}}$ 。一般在考慮可靠度的最佳化問題中只存在著非等式限制式，並沒有等式限制式，因爲若考慮不確定因素，則等式限制式幾乎不可能被滿足，更恰當的說法該說是，不確定因素並不適合出現在等式限制式當中，所以等式限制式並未出現在方程組(1.5)內。藉由方程組(1.5)可窺知考慮可靠度最佳化問題的架構。共有 n_d 個定值的設計變數， n_r 個隨機變數的設計變數，因此共有 $n=n_d+n_r$ 個設計變數，且有 m 個考慮可靠度的拘束條件。 [5,8,9]。

$$
\begin{aligned}
&\min_{\mathbf{d}=[\mathbf{x_d},\boldsymbol{\mu}_\mathbf{X}]} f(\mathbf{d}) \\
&\text{subject to} \;\; P_{\mathrm{f},j}=\Pr[g_j(\mathbf{d},\mathbf{U})>0]\le P_{\mathrm{f},j}^{\mathrm{allow}} \;\; j=1,\cdots,m \\
&\mathbf{d}^{\mathrm{LB}}\le\mathbf{d}\le\mathbf{d}^{\mathrm{UB}} \\
&\forall\mathbf{x_d}\in\Re^{n_d},\;\boldsymbol{\mu}_\mathbf{X}\in\Re^{n_r}
\end{aligned}
\tag{1.5}
$$

然而計算方程組(1.5)中的拘束條件的破壞機率 P_{f} 或可靠度 Re 皆是一項困難的任務，理論上破壞機率的計算需經由方程式(1.6)積分計算所得到。其中 $f_\mathbf{X}$ 爲隨機變數的聯合機率密度函數(joint probability density function)。若隨機變數之間爲線性獨立時，則聯合機率密度函數可改寫成各隨機變數的機率密度函數的積，如方程式(1.7)。若利用方程式(1.6)或方程式(1.7)直接積分，在實務上會遭遇到至少兩個問題，首先是聯合機率密度幾乎不可能得到，即使獲得了，多重積分依舊是個艱困的任務。此外當隨機變數不爲高斯分佈時，方程式積分值解析解不存在，更增添了計算可靠度的難度 [5]，因此有許多方法的問世希冀來嘗試解決可靠度估計的問題，文獻中可靠度的計算方法會在第2章中來做更詳細的介紹。

$$
P_{\mathrm{f}}=\Pr\left[g(\mathbf{X})>0\right]=\int\cdots\int_{g>0} f_\mathbf{X}\left(x_1,x_2,\cdots,x_n\right)dx_1dx_2\cdots dx_n \tag{1.6}
$$

$$
P_{\mathrm{f}}=\Pr\left[g(\mathbf{X})>0\right]=\int\cdots\int_{g>0} f_{X_1}f_{X_2}\cdots f_{X_n}dx_1dx_2\cdots dx_n \tag{1.7}
$$

1.2 研究動機與目的

1.2.1 研究動機

爲了處理可靠度最佳化問題，須著手處理可靠度與最佳化。 學者們試著使用可靠度指標(reliability index)的一階可靠度方法來(First Order Reliability Methods, FORM)估計可靠度 [10–12]，但以上方法只對隨機分佈爲高斯分佈時才較精準，因此有改善的一階方法處理非高斯分佈的方法問世 [13–15]。但一階方法仍然有估計誤差的產生，因此有二階可靠度方法(Second Order Reliability Methods, SORM)的產生 [16,17]，但也同時增加了計算的量與困難，並且在高度非線性的拘束條件中使用二階可靠度方法，但精確度還是不足。若使用樣本法(sampling)中常見的蒙地卡羅(Monte Carlo Simulation, MCS)可獲得精準的可靠度，但計算量卻相當大 [5]。因此若能提出一個減少樣本數的計算可靠度的方法，再配對最佳化演算法，即可解決可靠度最佳化的問題。

若估計可靠度方法選擇以樣本法做爲爲基礎，那配對最佳化演算法需選擇疊代次數少，效率高的演算法，來提升整體演算法的可行性。因此以 SQP (Sequential Quadratic Programming, SQP)做爲演算法的骨架，但 SQP 是一種需要梯度資訊的演算法，若要處理可靠度最佳化的問題就需計算機率梯度。因爲同樣情況下每次使用樣本法所計算的可靠度結果都不儘相同，這是樣本法的本質與特色，也就是使用樣本法得到的是一個擾動、跳動的結果，如圖1.2所示，圖1.2中的橫軸爲某一設計變數，縱軸爲破壞機率。若直接使用有限差分法來計算梯度，在此容易得到錯誤的訊息，而有限差分法(finite difference)其中一種向前有限差分法(forward difference)可寫成方程式(1.8)，其中$\hat{P}_{\mathrm{f},N}$ 代表著用 N 組樣本所估計出的破壞機率，$\Delta\mathbf{d}$ 爲變數的微小變化量。計算破壞機率梯度往往需要適當的微小變化量，才能計算出眞正相當應的破壞機率梯度，但適當的微小變化量卻難以制定，因而造成計算破壞機率梯度的結果錯誤。將圖1.2局部放大成圖1.3，更明顯的看出，對樣本法使用有限差分法很容易得到錯誤訊息，所以若演算法內包含樣本法與 SQP 勢必要解決計算梯度的問題，這也是本研究的動機之一。

$$\nabla_{\mathbf{d}}\hat{P}_{\mathrm{f}} \approx \lim_{\Delta\mathbf{d}\to 0} \frac{\hat{P}_{\mathrm{f},N}(\mathbf{d}+\Delta\mathbf{d}) - \hat{P}_{\mathrm{f},N}(\mathbf{d})}{\Delta\mathbf{d}} \tag{1.8}$$

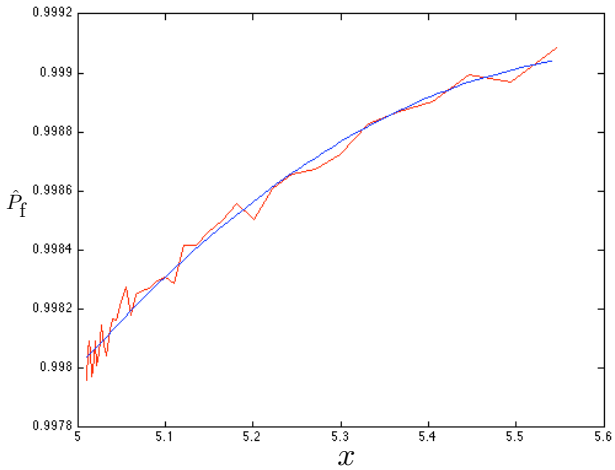

圖 1.2: 擾動破壞機率

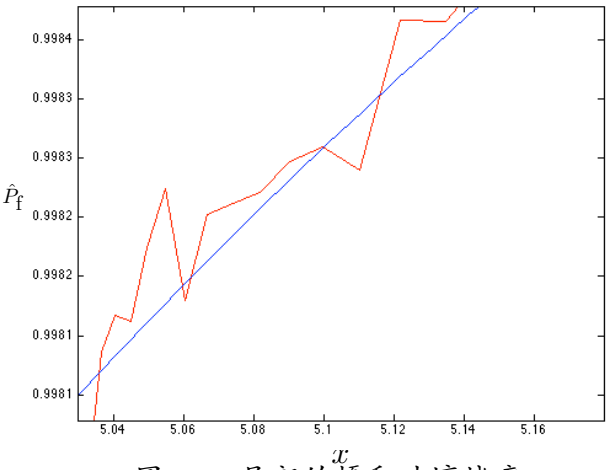

圖 1.3: 局部的擾動破壞機率

而另一動機爲在一般最佳化的演算法中，也面臨著如何制定懲罰函數與障礙函數，是否還有其它的處理方法，也成爲本文欲探討之重點。

1.2.2 研究目的

有鑑於上述目前文獻尚未解決的問題，本論文將提出一個以樣本法與 SQP 爲骨架的演算流程，來處理可靠度最佳化問題，並解決使用樣本法計算梯度所遭遇的困難，且提升演算法中樣本法的效率，來提升此演算法處理問題的速率與效率。整體來說，本論文所提出的演算法有以下特色

- 爲一針對可靠度最佳化問題所建構的梯度演算法
- 使用樣本法爲基礎來估計可靠度，並降低取樣的數量
- 融合 SQP 與樣本法來解決非線性可靠度拘束條件的設計問題
- 找出穩健的機率梯度計算方法以供 SQP 使用
- 使用一設計更新過濾器，“F-filter”，替代懲罰函數，並確保演算法的收斂

1.3 本文架構

本文共分六章。

- 第一章:前言
  介紹要處理的可靠度最佳設計問題及其相關背景，闡明研究動機與目的，並介紹論文架構。
- 第二章:文獻探討
  詮釋可靠度的定義，並針對現有的可靠度計算方法做一回顧，說明現存方法還有其不足之處與欲解決的問題。
- 第三章:取樣SQP演算法
  針對欲解決的問題，提出處理的方法與流程。
- 第四章:工程範例
  以一工程範例來說明提出的演算法深具可行性。
- 第五章:針對工程模擬函數的探討
  以不知數學形式的工程模擬拘束條件函數爲範例，來說明提出的演算法如何解決常見的電腦模擬問題。
- 第六章:結論與建議
  列出本研究之貢獻與結論，並提出相關建議，以作爲後續研究之參考。

第 二 章　文獻探討

大部份的工程量測或任何可以被觀察的現象均包含了不確定因素。不確定因素顧名思義是無法被預測，舉例來說，對某一現象在同一點重複量測或在不同狀態下量測均可能會有不同的量測結果，雖然有些量測結果出現的頻率會比其它結果高，但整體而言實際量值是無法預測的。在此文章中，可準確量測且預測的物理量稱之爲定性量值(deterministic)，反之則爲不確定因素(uncertainty)。

在現實的世界中，不確定因素環繞在我們的四周當中，舉例來說，任何產品都存在者公差，或更嚴格的說，沒有任何兩個產品是完全相同的。產品間的差異或是產品的公差，即是不確定因素的一種表現，一般來說，不確定因素的來源大致可分爲下列四種:

- 人爲因素:加工品質、量測精度、操作差異…等等
- 材料特性:楊式係數、浦松比、密度…等等
- 自然環境:溫度、濕度、氣壓、風速、地震、雷擊…等等
- 時間影響:疲勞、老化…等等

這些不確定因素都影響著產品的性能，因此若要確保產品的使用表現，勢必要將不確定因素考慮進來，因此估計可靠度或考量不確定因素所帶來的影響是可靠度最佳化與一般最佳化最大的不同。本章將針對不確定因素的模擬，可靠度的計算以及可靠度最佳化設計的方法及應用進行文獻回顧。

2.1　不確定因素模擬

爲了探討不確定因素所造成的影響，須先建構不確定因素的數學模型，隨機變數是文獻上最被廣泛應用的不確定因素數學模型。一個隨機變數的數學定義可由其機率密度函數(probability density function, PDF)或累積分布函數(cumulative distribution function, CDF)得知。文獻上有許多常用的分佈型態，工程師須對模擬的不確定因素有充份的認識與了解，才能進而選擇適合的機率分佈來模擬。隨機變數一般以大寫表示，在此以 X 表示，

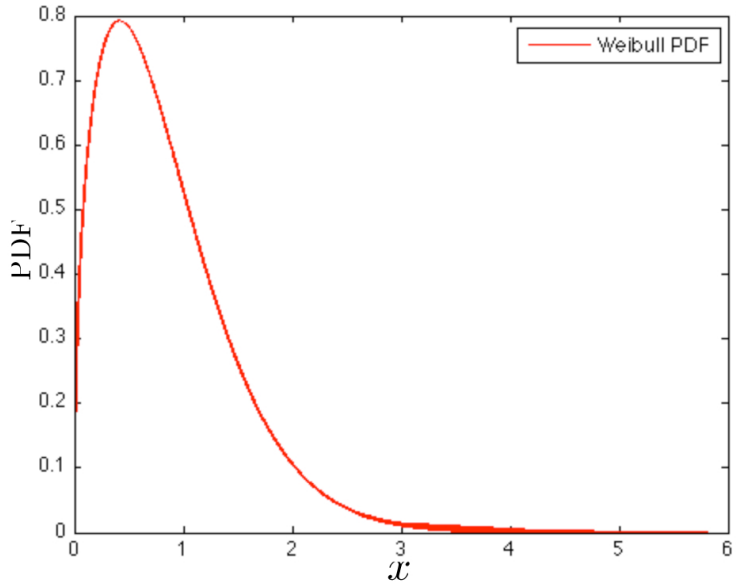

圖 2.1: 機率密度函數

又隨機變數當中可分爲連續(continuous)與離散(discrete)兩種型式。若 X 爲連續時，定義 f_X 爲機率密度函數，其意指爲隨機變數爲某一定值時的機率密度，如圖2.1所示，橫軸爲隨機變數，縱軸爲相對應的機率密度。若要計算 X 在某一區間 $[a,b]$ 內所發生的機率，等同於計算機率密度函數與 $X=a$ 、 $X=b$ 、橫軸所包圍的面積。數學形式可表示爲方程式(2.1)若 X 爲離散時，定義 f_X 爲機率質量函數(probability mass function, PMF)，其意義爲隨機變數爲某定值時的機率密度。 若要計算隨機變數小於某定值 e 的機率時可藉由累積分佈函數 F_X 來表示，如方程式(2.2)，因此可知累積分佈函數爲機率密度函數或機率質量函數積分的結果，所以機率密度函數與橫軸所之間的面積爲 1。

$$\int_a^b f_X(x)dx = \Pr\left[a \le X \le b\right] \tag{2.1}$$

$$F_X(x) = \Pr[X \le e] = \int_{-\infty}^{e} f_X dx \tag{2.2}$$

在此得知隨機變數的一些性質後，接著介紹最常見的隨機變數分佈，高斯分佈。 高斯分佈(Gaussian distribution)，亦稱常態分佈(normal distribution)，此分佈應用相當廣，不論在物理、化學、商業…等等領域中都相當常見其應用，是最常見的一種機率分佈之一。其特色爲左右對稱，高斯分佈的機率密度函數圖形如圖2.2，而數學形式可表示爲方程式(2.3)。其中 μ_X 、 σ_X 分別爲 X 的平均值與標準差，在使用上常見的標註爲 $X \sim N(\mu_X, \sigma_X^2)$ ，意指 X 爲高斯分佈，且其平均值爲 μ_X ，標準差爲 σ_X 。而高斯分佈的累積分佈函數可表示爲方程式(2.4)。爲了使用的方便，常將高斯分佈做標準化的一個動作，如方程式式(2.5)所示，在此

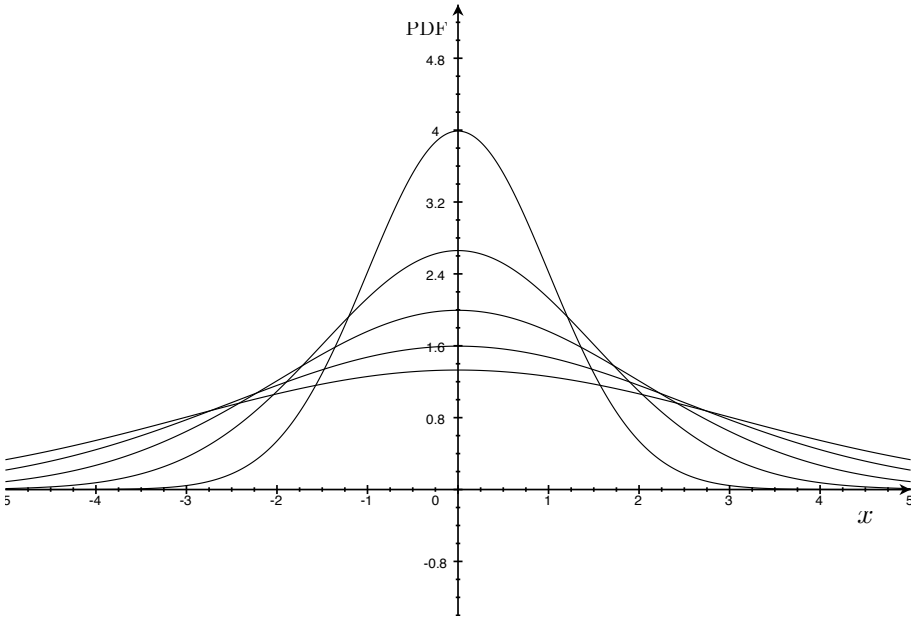

圖 2.2: 高斯分佈下的機率密度函數

U 皆爲平均值 0 ，標準差爲 1 的標準高斯分佈。

$$f(x) = \frac{1}{\sigma_X\sqrt{2\pi}} \exp\left[-\frac{1}{2}\left(\frac{x-\mu_X}{\sigma_X}\right)^2\right] \tag{2.3}$$

$$F_X(x) = \int_{-\infty}^{x} f_X dx \tag{2.4}$$

$$\mathbf{U} = \frac{\mathbf{X} - \mu_X}{\sigma_X} \tag{2.5}$$

而高斯分佈有許多的特色，除了對稱(symmetric)與鐘型(bell shaped)外，還有一些數學上的特質，若 $X \sim N(\mu_X, \sigma_X^2)$ 和 $Y \sim N(\mu_Y, \sigma_Y^2)$ 爲統計獨立的高斯隨機變數，則 V 也是常態分佈，表示如方程式(2.6)，隱含了高斯隨機變數經線性的轉換後仍爲高斯隨機變數。

$$V = X \pm Y \sim N(\mu_X + \mu_Y, \sigma_X^2 + \sigma_Y^2) \tag{2.6}$$

除了高斯分佈外，常見的隨機變數還有:對數分佈(lognormal distribution)、 韋式分佈(weibull distribution)、幾何分佈(geometric distribution)、浦松分佈 (poisson distribution)…等等。當了解一般常見的模擬不確定因素的方法之後，便是想使用此方法來估計產品或系統的可靠度。在下一小節中將會介紹如何估計可靠度。

2.2 可靠度估計

在對產品或系統估計可靠度之前，需先了解可靠度的定義，才能進一步的估計可靠度。可靠度可說是不確定因素下的產物，或是被影響後的結果。更嚴謹的定義爲:可靠度爲某裝置在特定的使用條件下。於規定期間內，毫無故障地發揮其特定功能的機率。其中“使用條件”、“規定期間”、“特定功能”、“機率”爲四個主要組成可靠度的因素。

- 使用條件

  意指產品與服務使用的前提或限制，如:特定的工作環境；溫度、溼度、壓力等等…，因爲在特定的使用環境下與操作下，對產品功能的可靠度才有評估的標準與依據。因此討論可靠度需在特定使用條件下始有意義，也解釋了一般商品的保固說明上，皆強調保固的範圍需在非人爲破壞與適合的工作環境下的原因。

- 規定期間

  時間，可說是另一種的不確定因素，在不同的時間下，產品的特性會改變，常見的爲零件老化、疲勞。因此，明定產品的標準使用時間，更有助於可靠度的分析，與使用者的操作，但在本文中並不對時間的影響作討論。

- 特定功能

  可靠度中很重要的一環即是“明確”定義產品功能，再藉由產品功能的定義來判斷產品是否破壞。“明確”的定義是指任何情況下依其準則來判斷有相同的結果。在本文中若產品失去功能則稱“破壞”(failure)，反之稱爲“安全”(safe)。

- 機率

  在可靠度的分析中，此項是分析的結果。產品穩定的程度藉由機率來作爲指標也就是可靠度的單位

由上述的定義，可清楚知道，如何判斷產品與系統的狀態是安全或是破壞。再利用機率的概念來闡述安全或破壞的情形。因此衡量破壞情形的稱爲破壞機率(probability of failure)。

可靠度的估計往往是針對某一特定產品或系統的某一情形下做估計。若以某一模擬來取代該可靠度試驗，而該模擬會融入量化的不確定因素，若該模擬以數學函數形式來表示，則會用函數來表達該產品性能，變數來顯示不確定因素及建構產品性能的因子。一般會以隨機

變數來模擬不確因素，而該組的隨機變數稱爲估計變數或是估計點。在第2.1節中，介紹了模擬不確定因素，隨機變數及其特性後，在此節中就是要介紹一般常見的可靠度估計方法。

在此介紹常見的計算方法，將其分爲三類

- 一階可靠度方法(First Order Reliability Methods, FORM)
  - 一階二次可靠度方法(First Order Second Moment, FOSM)
  - 進階一階二次可靠度方法(Advanced First Order Second Moment, AFOSM)
- 二階可靠度方法(Second Order Reliability Methods, SORM)
  - 二階可靠度方法(Second Order Reliability Method, SORM)
- 樣本法(Sampling)
  - 蒙地卡羅法(Monte Carlo Simulation, MCS)
  - 減少變異的取點技法 (Variance Reduction Techniques, VRT)
    - 重要取點法(Important Sampling, IS)
    - 半徑取點法(Radius-based Important sampling, RIS)
    - 條件期望值(Conditional Expectattiono, CE)

2.2.1 一階二次可靠度方法

在工程領域中，往往希望以低成本、少計算量的計算方法來獲得可靠度或破壞機率，因此一階二次可靠度方法(First Order Second Moment, FOSM) [10]因而誕生，此法又名平均值一階二次可靠度方法(mean value first order second moment, MV-FOSM)。此法特色爲利用泰勒展開式(Taylor series)，在估計變數的平均值上做一階展開，如方程式(2.7)，以及使用隨機變數的兩個矩量平均值、標準差來作可靠度的估計，這些特色也反應在其命名上。一階二次可靠度方法步驟如下:

- 第一步驟:

  對某一拘束條件 $g_j(\mathbf{X})$ 做泰勒一階展開，如方程式(2.7)，其用意爲將原拘束條件線性化轉化爲 $g'_j(\mathbf{X})$ ，期許用 $g'_j(\mathbf{X})$ 取代 $g_j(\mathbf{X})$ ，將原函數轉換爲線性的一階泰勒展開式後， $g'_j(\mathbf{X})$ 的平均值與標準差都皆可由簡單的算式獲得。

$$g'_j(\mathbf{X}) = g(\boldsymbol{\mu_X}) + \sum_{i=1}^{n} \frac{\partial g}{\partial x_i}(X_i - \mu_{X_i}) \tag{2.7}$$

- 第二步驟:

  計算 $g'_j(\mathbf{X})$ 的平均值與標準差， 因爲 $g_j(\mathbf{X})$ 被轉換 $g'_j(\mathbf{X})$ ，而 $g'_j(\mathbf{X})$ 爲線性函數，可視爲一個高斯的隨機變數，可輕易求得平均值、標準差。

$$g'_j(\mathbf{X}) \sim N(\mu_{g'_j}, \sigma^2_{g'_j}) \quad \text{在此} \begin{cases} \mu_{g'_j} = g(\mu_{X_1}, \mu_{X_2}, \cdots, \mu_{X_n}) \\ \sigma^2_{g'_j} = \sum_{i=1}^{n} \left(\frac{\partial g}{\partial x_i}\right)^2 \sigma^2_{X_i} \end{cases} \tag{2.8}$$

- 第三步驟:

  利用平均值、標準差與累積分佈函數來求得 $g'_j(\mathbf{X})$ 的可靠度。再以 $g'_j(\mathbf{X})$ 的可靠度來估於於原拘束條件 $g_j(\mathbf{X})$ 的可靠度，在此 Φ 爲標準化高斯隨機變數的累積分佈函數(平均值爲0標準差爲1)。在此 β 定義爲可靠度指標或安全指標(reliability index or safety index)。

$$\hat{P}_{\mathrm{Re}} = 1 - \Pr[g'_j(\mathbf{X}) > 0] = 1 - \Phi(\frac{\mu_{g'_j} - 0}{\sigma_{g'_j}}) = 1 - \Phi(\beta) \tag{2.9}$$

$$\beta = \frac{\mu_{g'_j}}{\sigma_{g'_j}} = \frac{g'_j - 0}{\sqrt{\sum_{i=1}^{n} \left(\frac{\partial g}{\partial x_i}\right)^2 \sigma^2_{X_i}}} \tag{2.10}$$

依上面的步驟可發現一階二次可靠度方法，可很快的計算出可靠度， 但使用一階二次可靠度方法只有在少數的例子可以計算出眞正的破壞機率， $g_j(\mathbf{X})$ 須爲線性，且隨機變數的分佈須爲獨立(statistically independent)的高斯分佈。然而在工程上的問題大部份都不爲線性，當 $g_j(\mathbf{X})$ 不爲線性時，估算的可靠度會有相當的誤差產生 [10]。一階二次可靠度方法有些較爲值得改進的地方，因爲此法不論隨機變數分佈爲何，完全忽略了當隨機變數分佈的資訊是已知時所帶來的影響。而此方法在隨機變數爲高斯分佈時有較佳的效果

2.2.2 進階一階二次可靠度方法

進階一階二次可靠度方法(Advanced First Order Second Moment, AFOSM)是一種適用於隨機變數爲高斯分佈的計算方法，由 Hasofer,A. 與 Lind,N.在 1974 年所提出 [12]，也被簡稱爲 H-L 法。進階一階二次可靠度方法爲目前常用的可靠度估計方法之一，其有不錯的準確度，使用的成本與時間也被工業界所接受。 此法可分爲下列三個步驟:

- 第一步驟:
  將隨機變數轉換爲至標準空間內，即隨機變數標準化，如方程式(2.11)所示。

$$\mathbf{u} = \frac{\mathbf{x_i} - \boldsymbol{\mu}_{\mathbf{X_i}}}{\boldsymbol{\sigma}_{\mathbf{X_i}}} \tag{2.11}$$

- 第二步驟:
  找到標準空間下拘束條件的極限狀態(limit state, $g_j(\mathbf{X}) = 0$)上的一點，此點與標準空間的原點的距離須爲最短。因此這形成一個最佳化問題，如方程式(2.12)所示，滿足上述最佳化問題所找到的最佳解，稱爲最大可能破壞點(most probable point, MPP)，Rackwitz在 1976 年提出了一套流程來處理以上的最佳化問題 [11] 。此作法的原因很直觀，在做可靠度分析時，最被關心的就是何時發生第一次破壞，也就是在拘束條件上離估計點最近的點，就是上述的最大可能破壞點。因此最容易發生破壞的情形就發生在估測點與距離拘束條件最近的點，也是最大可能破壞點命名的由來。

$$\begin{aligned} &\min_{\mathbf{u}} \mathbf{u}^T\mathbf{u} \\ &\text{subject to} \quad g_j(\mathbf{u}) = 0 \end{aligned} \tag{2.12}$$

- 第三步驟:
  其中可靠度指標 $\beta = \sqrt{\mathbf{u}^T\mathbf{u}}$ ，利用方程式(2.13)可計算破壞機率，由方程式(2.13)，可得一個基本概念就是，若估計變數與極限狀態的拘束條件距離(β)愈遠，則可靠度愈高，反之亦然。

$$\hat{P}_{\mathrm{f}} = \Phi(-\beta) = 1 - \Phi(\beta) \tag{2.13}$$

在此如同一階二次可靠度方法中使用可靠度指標來憑估可靠度，若問題中的 $g_j(\mathbf{X})$ 爲線性的且隨機變數分佈爲獨立的高斯分佈，那一階二次可靠度方法與進階一階二次可靠度方法

會得到相同的結果，且是眞正的破壞機率。然而在非線性時，進階一階二次可靠度方法因經過一個最佳化過程來找尋最大可能破壞點，讓計算出的可靠度誤差不至於如一階二次可靠度方法般的這麼大，但相對的須付出更多的計算成本。此外進階一階二次可靠度方法在使用上如之前所述，隨機變數分佈應爲高斯分佈，爲了克服此缺點，Rackwitz 和 Fidssler在 1978 年 [15]、Hohenbichler 和 Rackwitz在 1981 年 [13]以及 Chen 與 Lind 在 1983 年 [14]分別提出一些方法試著塡補此點不足。

一般將只考慮函數一階特色的方法歸類於一階可靠度方法(FORM)，或是用可靠度指標來估計的方法都概括稱爲一階可靠度方法 [18]。在的文獻中只要使用一階可靠度方法，皆會面臨到拘束條件爲非線性，所造成的誤差與造成找尋最大可能破壞點的誤差問題 [19]。爲了解決此問題，在下一小節將介紹二階可靠度方法來改善此問題。

2.2.3 二階可靠度方法

有鑑於使用一階可靠度方法，線性可獲得準確的可靠度值，非線性的結果會產生誤差。但在工程領域中，遭遇到的問題大多爲非線性，一階可靠度方法(FORM)並不足以滿足工程師的需求，因此有二階可靠度方法 (second order reliability methods, SORM)的誕生來幫助工程師解決問題。 二階可靠度方法的概念最先是由 Fiessler 等人在 1979 年首先提出 [16]，之後 Breitung 等人在 1984 年提出的二階可靠度方法是目前較爲常見的版本 [17]。二階可靠度方法是藉由考慮極限狀態下拘束條件的曲率，來改善一階可靠度方法，也就是需對拘束條件作二階泰勒展開。如方程式(2.14)，二階可靠度方法可分爲下列幾個步驟:

$$g_j'(\mathbf{X}) = g(\mu_X) + \sum_{i=1}^{n} \frac{\partial g}{\partial x_i}(x_i - \mu_{X_i}) + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \frac{\partial^2 g}{\partial x_i \partial x_j}(x_i - \mu_{X_i})(x_j - \mu_{X_j}) \tag{2.14}$$

- 第一步驟:
  將隨機變數轉換爲至標準空間內，即隨機變數標準化，如方程式(2.11)所示。

- 第二步驟:
  計算 B_0 與 A ，其中 B 爲 B_0 的正交矩陣(orthogonal matrix)， D 爲拘束條件的Hessian矩陣，拘束條件函數的局部主曲率(principal curvatures)爲 A 矩陣的特微值 κ (eigenvalue)。

$$B_0 = \begin{bmatrix} 1 & 0 & . & . & . & 0 \\ 0 & 1 & 0 & . & . & 0 \\ . & . & . & . & . & 0 \\ . & . & . & . & 1 & 0 \\ \frac{\partial g}{\partial u_1} & \frac{\partial g}{\partial u_2} & . & . & . & \frac{\partial g}{\partial u_n} \end{bmatrix}$$

$$A = \frac{B^T D B}{||\nabla g(\mathbf{u})||} \tag{2.15}$$

- 第三步驟:

  估計破壞機率 $\hat{P}_{\mathrm{f}}$ ，其中 β 爲使用一階可靠度方法所估計而來的，最終可獲得估計破壞機率 $\hat{P}_{\mathrm{f}}$。

$$\hat{P}_{\mathrm{f}} = \Phi(\beta) \prod_{i=1}^{n} (1 + \beta\kappa_i)^{\frac{-1}{2}} \tag{2.16}$$

因爲Breitung 的方法是使用拋物線來近似 [17]，不是二次函數的通式型態，因此 Breitung 的計算方式只有在可靠度指標(β值)較大時才會準確，當可靠度指標值較小時會造成不準確的計算結果 [17]。 Tvedt 等人在 1990 年提出一個替代方法來試著解決這樣的問題 [20]。由此可知以上所提的近似方法;一階可靠度方法、二階可靠度方法都有其不足之處，因此在下面的小節將介紹取樣(sampling)的技法，來計算可靠度。

2.2.4 蒙地卡羅法

蒙地卡羅法(monte carlo simulation, MCS)是一般很常見的可靠度估算法 [5]，運用的層面也相當廣，從人文至科學都不難發現其蹤跡，例如文獻中的 [21–23]。

蒙地卡羅法計算相當簡單，就是問題中的每一個隨機變數皆被大量取樣，來盡量表現該隨機變數的特性，因此被取樣的樣本應符合該隨機變數出現的機率密度。所以蒙地卡羅法是以統計學爲基礎，來決定結果的一種方法，因此，此方法的成敗關鍵就在於是否能完整的將隨機變數的特性充份展現。在此考慮某一拘束條件$g(\mathbf{X})$，常見的數學表示如方程式(2.17)，$I_g(\mathbf{x})$ 爲指標函數，判斷是否破壞，其中 $\mathbf{x}$ (realizations)是從母體 $\mathbf{X}$ 所取出的樣本， $\hat{P}_{\mathrm{f}}$ 爲

用蒙地卡羅法所計算出的破壞機率，N 爲總樣本數，N_f 爲破壞的總數，當方程式(2.18)中的 $N \to \infty$ 其意義如同於方程式(1.6)中積分符號的概念，在此寫成方程式(2.19)，因此當 $N \to \infty$ 時，方程式(2.18)中的 估計破壞機率值 $\hat{P}_{\mathrm{f}}$ 會等同於方程式(2.19)的眞實破壞機率值 P_{f} 。

$$I_g(\mathbf{x}) = \begin{cases} 1, & \text{if } g(\mathbf{x}) > 0 \\ 0, & \text{otherwise.} \end{cases} \tag{2.17}$$

$$\hat{P}_{\mathrm{f}} = \frac{1}{N}\sum_{i=1}^{N} I_g(\mathbf{x}) = \frac{N_f}{N} \tag{2.18}$$

$$P_{\mathrm{f}} = \int_{g(\mathbf{x})>0} I_g(\mathbf{x}) f_X(\mathbf{x}) d\mathbf{x} \tag{2.19}$$

實行蒙地卡羅大致可分爲下列五個主要步驟:

1. 定義問題中的隨機變數
2. 量化各隨機變數的機率密度函數(PDF or PMF)及其相關的參數。
3. 根據隨機變數的分佈型態，產生樣本
4. 將產生的樣本，使用至函數或實驗中，並收集、整理結果。
5. 將多組結果轉換成可靠度。

蒙地卡羅是很直觀的一種的技法，它的準確性也相當的好，但爲了完整的將隨機變數的特性充份展現，需要相當大量的樣本數。此外若估計變數與此拘束條件間的最近距離相當遠，因此若估算可靠度時直接用原分佈取樣，需取相當大量的樣本才能獲得所需的結果，若能在較靠近破壞的區域上取樣，可減少樣本數，可參考文獻 [24]。如圖2.3，因爲估計變數與極限狀態的拘束條件的距離甚遠，樣本數不足的蒙地卡羅會得到偏差的結果，得到可靠度百分之百的訊息，但若增加蒙地卡羅的樣本數，如圖2.4，樣本數增加至十萬組時，可估計出非常接近眞實情形的可靠度約 99%，因此蒙地卡羅雖準確，但效率卻不高，造成實際應用上的困難。

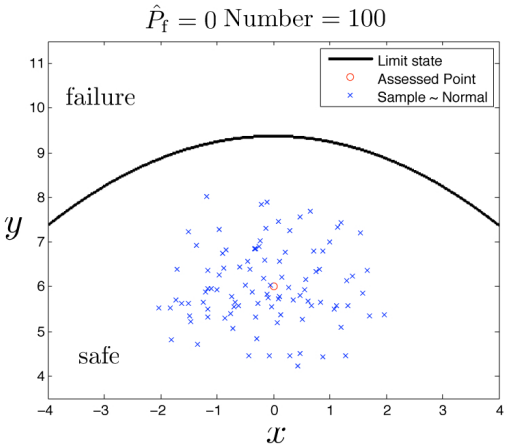

圖 2.3: 樣本數少造成估計誤差

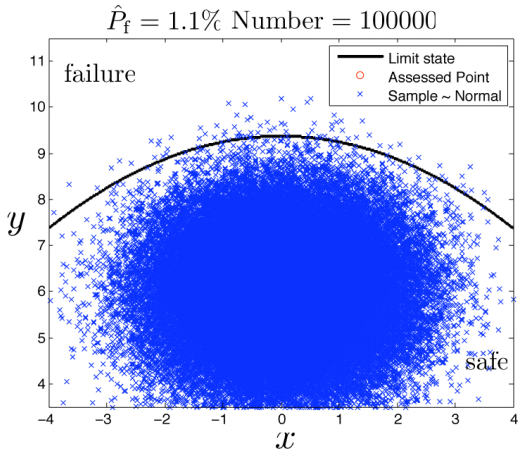

圖 2.4: 樣本數多減少估計誤差

爲了提升效率與應用層面，必定要降低取樣的數量，因此減少變異的取點技法(variance reduction techniques, VRT)會在下面的章節做介紹，一般 VRT 常見的方法有重要取點法(important sampling, IS)、半徑取點法(radius-based important sampling, RIS)、條件期望值(Conditional Expectation, CE)…等等。儘管減少變異的取點技法可改善效率，但同時的也增加計算的難度。在下面一小節中將介紹減少變異的取點技法的概念與方法。

2.2.5 減少變異的取點技法

在一般的取點方法中，全部的隨機變數，隨機產生多組樣本後，再憑藉著樣本的狀態爲安全或破壞並與全部取樣的比例來決定機率值的大小。若某一個拘束條件眞實的破壞機率是一個極小值，往往需要極大量的樣本數來估計，才能計算出一個含有可接受的誤差的結果，可參考文獻 [24]。變異量的含義爲估計的破壞機率與眞實的破壞機率的差平方的期望值，如方程式(2.20)，其中 z 爲用某一樣本數所估計的次數，當變異量愈大就代表結果愈不精準，在統計理論當中，樣本數愈多，則變異會隨著變小，變異量與樣本呈現互相拉扯的情形，可參考文獻 [25]，因此往往需要使用大量的樣本來降低變異。大量樣本的使用造成效率的低落與成本的增加，爲了達到效率的提升，樣本數需要有顯著的下降，或者是降低變異且在不增加樣本數的前提下。因此減少變異的方法就是減少樣本數的方法。

$$\mathrm{Var}(P_f) = \frac{\sum_{i=1}^{z}(\hat{P}_{f,i} - P_f)^2}{z} \tag{2.20}$$

變異降低的方法是在某些特定的情況或假設下所可以使用的，換句話說就是必須先已知

某些特定的額外資訊，才可使用的方法。一般來說是無法事先預期變異的減少量，也就是說經減少變異的取點技法估計前，無法事先得知該計算結果的變異量，可參考文獻 [25]。此說法也隱含著，估計前無法訂定一個適合的樣本數，來估計可靠度，只能估計完，事後再來判斷這樣的結果是否可被使用者所接受。這樣的問題不只在減少變異的取點技法中會遇到，事實上大部份只要是使用樣本的方法都會遭遇到訂定樣本數的問題，或說是變異的問題。

將變異降低的方法主要有幾種改善的概念，第一種是試圖降低變異量，例如在蒙地卡羅中，提供的僅僅是 0 與 1 的資訊，因此改善的概念則是改善只提供 0 與 1 資訊的問題，因爲用 0 與 1 代表安全與破壞，但事實上即使破壞或安全的點也是還有程度的差別，例如相同不違反拘束條件的兩個點，距離極限狀態的拘束條件較近的點，則其可靠度較低，反之亦然。因此若只提供 0 與 1 的資訊似乎顯的太少。因此若有一估計方法可提供 0 至 1 之間的資訊來作爲機率值，不僅提供更多元的資訊內容外，其變異也可降低，因爲每組樣本估計的結果會較爲近似，則此估計方法就可用較少的樣本數來估計。而另一些方法，則是經由特別的分析方法，來達到減少變異的目的。

減少變異的取點技法可提升估計可靠度的效率與精確度，並使用較少的樣本數，但也同時增加了每次取樣與計算的難度。一般使用減少變異的取點技法的技巧幾乎不可能預先知道所使用的方法，可以造成多少的變異減少，可參考文獻 [25]。

2.2.6 重要取點法

重要取點(importance sampling, IS)的基本概念著重於取樣的區域，取樣在認爲重要的區域，而那些區域主要與破壞機率有關。因此可用較少的樣本來得到可靠度的結果，所以如何獲得哪些才是重要區域的資訊變的極爲關鍵。重要取點法的計算推導如下，首先將破壞機率方程式(1.6)，加入指標函數 I_g ，可改寫成方程式(2.21)。

$$P_{\mathrm{f}} = \int_{g(\mathbf{x})>0} I_g(\mathbf{x}) f_{\mathbf{X}}(\mathbf{x}) d\mathbf{x} \tag{2.21}$$

在此定義新區域的分佈機率密度函數爲 $f'_{\mathbf{X}}(\mathbf{x})$ ，利用新機率密度函數可得空間內的樣本。明顯的，新機率密度函數 $f'_{\mathbf{X}}(\mathbf{x})$ 須爲已知，可依使用者需要來決定。而破壞機率可再改寫爲方程式(2.22)，方程式(2.22)是由方程式(2.21)同時乘上、除以 $f'_{\mathbf{X}}(\mathbf{x})$ 。所以若以取樣本方法計算時可改寫方程式(2.22)，改寫方法如同於第2.2.4小節中的方程式(2.19)等效於方程式(2.18)的過程。因此最後可得如方程式(2.23)結果， N爲總樣本數 ， $\mathbf{x}_i$ 爲第 i 組樣本的隨機變數值 $(x_1, x_2, \cdots, x_n)_i$，$n$ 爲隨機變數的組數。因此可知重要取點法，具有轉換隨機變數分佈的能力，轉換隨機變數分佈的影響，可細分爲下列兩種:改變隨機變數的參數與轉換隨機變數的分佈。

$$P_{\mathrm{f}} = \int_{g(\mathbf{x})>0} \left[I_g(\mathbf{x}) \frac{f_{\mathbf{X}}(\mathbf{x})}{f_{\mathbf{X}'}(\mathbf{x})} \right] f'_{\mathbf{X}}(\mathbf{x}) d\mathbf{x} \tag{2.22}$$

$$\hat{P}_{\mathrm{f}} = \frac{1}{N} \sum_{i=1}^{N} I_g(\mathbf{x_i}) \frac{f_{\mathbf{x}}(\mathbf{x_i})}{f'_{\mathbf{X}}(\mathbf{x_i})} \tag{2.23}$$

- 改變隨機變數的參數

  舉例來說:原來的隨機變數分佈爲高斯分佈，新的隨機變數分佈還是高斯分佈，但新的高斯分佈可以有不同的平均值。由此例就可說明重要取點法，可改變取樣的區域，因此可取樣在認爲較重要的區域，來減少取樣的數目。如圖2.5所示，$\circ$ 代表原取樣的中心，爲高斯分佈的中心， $\times$ 爲以原中心所取的高斯隨機變數樣本，$\diamond$ 代表移動後的中心，$+$ 爲移動後的中心所取的高斯隨機變數樣本，而接近破壞的區域就是較重要的區域。在一般的考慮不確定因素的最佳化問題，所要求的可靠都相當相當的高，這也暗示了“可靠度估計點與拘束條件靠最近的點有相當遠的距離”的這個現像很常發生在可靠度最佳化問題當中，因此使得找到重要取點的區域，變得極爲重要。但如何事先

判斷何爲重要區域，也成爲使用重要取點法上一個困難。若欲利用此性質，需對問題特色有相當程度的了解，取決於工程師對問題的定義與認知。

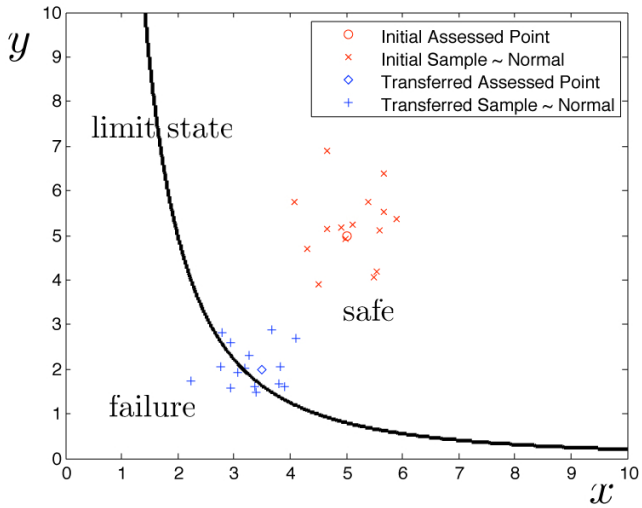

圖 2.5: 重要取點法改變隨機變數的參數

- 轉換隨機變數的分佈

  除了改變隨機變數的參數外，重要取點還可改變隨機變數的分佈，如同上述的例子，原隨機變數分佈爲高斯分佈，新的隨機變數分佈可轉換爲均勻分佈或是其它分佈。如圖2.6所示，Harbitz 在 1986 年 [26]與 Karamchandani 等人在 1989 年 [27]作過關於自定的新區域分佈的機率密度函數($f'_{\mathbf{X}}(\mathbf{x})$)的一些探討。

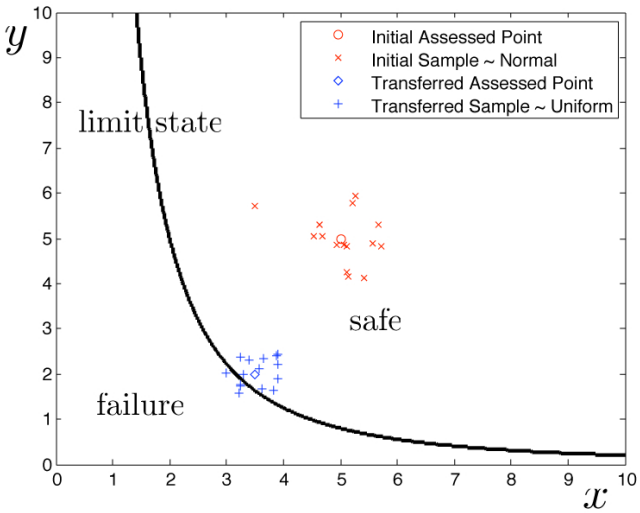

圖 2.6: 重要取點法轉換隨機變數的分佈

2.2.7 條件期望值

在條件期望值(conditional expectation, CE)中，全部的隨機變數都須被模擬，除了一個隨機變數例外，稱爲控制變數(control variable)。一般來說選擇控制變數的條件爲具有較大的變量(variability) [5]，且已知其累積分佈函數(CDF)爲一已知函數, 該函數由除了控制變數以外的隨機變數所組成，並且控制變數與該函數內的其餘隨機變數彼此之間相互獨立(independent)。而產生的變異量因扣除了控制變數的隨機擾動因而有了降低，可參考文獻 [24,25]。實行步驟如下

- 步驟一：
  選擇控制變數 X_1 ，其中 $\mathbf{X} = [X_1, \bar{\mathbf{X}}]$ ， $\bar{\mathbf{X}} = [X_2, X_3, \cdots, X_n]$
- 步驟二：
  重新改寫方程式，可改寫成方程式(2.24)或方程式(2.25)

$$P_{\mathrm{f}} = \Pr[\ \ X_1 < h_j\left(\bar{\mathbf{X}}\right)] \tag{2.24}$$

或

$$P_{\mathrm{f}} = \Pr[\ \ X_1 > h_j\left(\bar{\mathbf{X}}\right)] \tag{2.25}$$

- 步驟三：
  產生除了控制變數以外的隨機變數 $\bar{\mathbf{x}}$($\bar{\mathbf{x}}$ 爲由母體 $\bar{\mathbf{X}}$ 所取出的樣本)

$$\hat{P}_{\mathrm{f}_i} = \Pr[\ \ X_1 < h_j\left(\bar{\mathbf{x}}_i\right)]$$

或

$$\hat{P}_{\mathrm{f}_i} = \Pr[\ \ X_1 > h_j\left(\bar{\mathbf{x}}_i\right)] \tag{2.26}$$

- 步驟四：
  重複步驟三 N 次後，計算 $\hat{\mathbf{P}}_{\mathrm{f_i}}$ 的平均值，方程式(2.27)，而變異數可由方程式(2.28)來獲得

$$\hat{P}_{\mathrm{f}} = \frac{\sum_{i=1}^{N} \hat{P}_{\mathrm{f}_i}}{N} \tag{2.27}$$

$$\mathrm{Var}(\hat{P}_{\mathrm{f}}) = \frac{\sum_{i=1}^{N}(\hat{P}_{\mathrm{f}_i} - \hat{P}_{\mathrm{f}})^2}{N(N-1)} \tag{2.28}$$

條件期望值是一種很常被使用來估計結構的可靠度的方法，如文獻 [28–30]，此方法有很好的效率，並且在較少的樣本下提供了足夠精確的可靠度值，見文獻 [28,29]。

2.2.8 可靠度方法比較

上段文章中，介紹了許多可靠度計算方法，因此在這做一整理，大致可分爲三類

- 一階可靠度方法 (FORM)
  一階可靠度方法方法就是考慮線性化後的拘束條件並利用統計中的兩個矩量，來估計可靠度，效率高、計算量少，但因只考慮一階展開的結果，使得估計出的結果還有不足的地方。因此進階一階二次可靠度方法爲了減低估計的誤差，在估計時可靠度引進了最佳化的概念，來降低誤差。上述的兩種估計方法，都在隨機變數分佈爲高斯分佈時，有較準確的結果，所以之後有許多學者提出解決當隨機變數分佈爲非高斯分佈時問題的方法。但因爲一階可靠度方法是將拘束條件做一階展開，其中一階二次可靠度方法是在估計變數上展開，而進階一階二次可靠度方法則是在最大可能破壞點(MPP)上展開，此點滿足拘束條件的極限狀態並且與估計點有最近的距離，因此當遇到非線性的拘束條件時，進階一階二次可靠度方法的計算誤差與一階二次可靠度方法相較之下少了許多。雖然進階一階二次可靠度方法已改善一階二次可靠度方法的計算誤差 ，但還是有誤差產生尤其當拘束條件爲高度非線性時，誤差仍是顯著，二階可靠度方法也因此而被發展。

- 二階可靠度方法 (SORM)
  二階可靠度方法，承接了一階可靠度部分的概念，並增加考慮拘束條件極限狀態下的曲率來做修正可靠度指標(reliability index)的依據，考慮曲率須將拘束條件做二階展開，這也是此類方法命名的由來。 計算量比起一階可靠度方法增加了不少，而計算出的結果有比一階可靠度更爲改善，但若是拘束條件爲高度非線性時，或說是該拘束條件以泰勒展開式展開後，其中二次以上的項次不可被忽略時，就會產生重要(significance)的誤差，但在工程領域當中拘束條件往往不是二階展開就足以取代的，

表 2.1: 可靠度方法比較

方法	FORM	SORM	Sampling	Sampling with VRT
優點	計算量少	結果較FORM精確	結果精確	結果精確 計算量較MCS少
缺點	結果不夠精確	結果不夠精確 計算量較FORM增加	計算量大	增加計算的難度

或說不知拘束條件需要幾階的展開才足以代表原函數，因此二階可靠度方法還是有不足的地方。

- 樣本法(Sampling)

  最常見的蒙地卡羅法可說是一般最準確的可靠度估計方法，其可靠度估算結果被眾人視爲比較的準繩，但爲了將不確定因素的特色完全展現，需要大量的樣本來評估，因此最大的缺點就是計算量過大，但若遇到較昂貴或費時的模擬時，大量的取樣就成爲蒙地卡羅法實行的一大阻礙，可參考文獻 [5,31,32]。 因此爲了保持樣本法精確性，並且降低樣本數的用量，因此有減少變異的取點技法(VRT)的問世，常見的方法如:重要取點(IS)、半徑取點法(RIS)。

  重要取點法(IS)著重於取樣的範圍與取樣的分佈，見文獻 [26]，但重要取點法(IS)中的取樣範圍不易定義，因此有了將可靠度指標(reliability index)概念融入取點的方法，也就是半徑取點法(RIS)，可參考文獻 [26]。但利用可靠度指標(reliability index)還是無法有效的判斷取樣範圍，因此有了重要調整取點法(adaptive important sampling, AIS)的出現，可參考文獻 [33]，重要調整取點法融入了半徑取點法的概念並可調整樣本的大小根據之前的樣本資訊，換句話說就是多次取樣來得到最終所要的資訊，在此之後更有考慮曲率的重要調整取法方法出現，可參考文獻 [34] 。但使用減少變異的取點技法往往需要一些額外的資訊，而這些資訊卻是不太容易在進行可靠度估計前所可以取得的。

用表2.1來總結上述可靠度方法的特色，可知拘束條件的非線性程度對估計產生的誤差有顯著的影響，儘管還是有些方法可以處理高度非線性的問題，但往往是相當花費成本的或是需要額外的資訊幫助。

2.3 可靠度最佳化之方法與發展

解決可靠度最佳化問題都需要處理兩個問題，最佳化與可靠度的計算。可靠度中常見的一些方法往往都需要找尋最大可能破壞點，而找尋最大可能破壞點本身就是一個最佳化過程，因此解決可靠度最佳化問題形成一個雙迴圈(double loop)的演算內容，見文獻 [35]。雙迴圈的構成分別是由外迴圈的最佳化以及內迴圈計算可靠度的最佳化過程所建構而成的。常見的方法有可靠度指標法(reliability index approach ,RIA)，其內迴圈所使用的估計可靠度方法爲進階一階可靠度方法爲 H-L 法。因爲可靠度指標法所需的計算過高，因此又有學者提出以雙迴圈爲基礎的另一方法性能量測法(performance measure approach, PMA)。性能量測法比起可靠度指標法來得更有效率，可參考文獻 [35,36]，因爲性能量測法在內迴圈的計算量比起可靠度指標法少了許多。但因爲雙迴圈的處理過程爲迴圈中還有迴圈，造成計算的成本還是偏高，因此更有學者，將找尋最大可能破壞點的最佳化的條件(optimality condition)融入在演算法中，提出使用單一迴圈(single loop)的方法，可參考文獻 [37,38]。單一迴圈的方法因融入了最佳化條件，改變了內迴圈的形式，轉換成如同外迴圈增加了一個等式的限制式的最佳化問題，因此內迴圈如同變相的消失，形成單一迴圈的過程，大大提升了效率並保有類似雙迴圈的精確性。

2.4 待解決的問題

隨著科技的發展一日千里，使用的產品日益精良，要求也越趨嚴格，因此所要求的可靠度也越趨精確，因此希望借重樣本法來計算可靠度，以樣本法爲基礎來估算可靠度，並且儘可能的降低樣本數目，來減少成本的支出。

處理可靠度最佳化的問題需將最佳化演算法與可靠度估計做一個巧妙的結合。若以較花費成本的樣本法來估計可靠度，則與之搭配的最佳化演算法，需具備計算疊代次數較少的特色，若選擇計算次數較多的演算法，會造成整體在處理考慮可靠度的最佳化問題過程中，花費的成本與計算次數過於龐大，造成眞正實行上的困難，因此選擇以梯度爲基礎的循序二次規劃演算法(sequential quadratic programming, SQP)爲骨幹。 但若想處理考慮可靠度的最佳化問題，還有一個艱難的困難要面對，就是若使用 SQP 爲演算法，需提供梯度資訊，然而

樣本法所計算出的破壞機率爲一個帶有擾動的結果如圖1.2與圖1.3所示，若使用有限差分法來計算破壞機率梯度，計算出的結果會有錯誤。

另一難題在一般最佳化問題中常會遇到的，就是在演算法中的疊代過程所找到的設計變數造成拘束條件違反。在文獻中最常見的處理方式爲使用“懲罰函數”(penalty function)，可參考文獻 [1]，其方法簡單的說就是將目標函數值及拘束條件形成一如方程式(2.29)的懲罰函數(penalty function)。若最佳化問題型式爲全負型態(negative null form)，則懲罰項 $F(\mathbf{d})$ 爲非負函數在 $\mathbf{d}$ 的定義域中，反之亦然，並且若設計變數不使拘束條件違反，則 $F(\mathbf{d}) = 0$。但使用懲罰函數有其困難的地方，就是如何選擇懲罰函數 $F(\mathbf{d})$ ，一般令懲罰函數等同於拘束條件的違反量值的平方，因此懲罰項可表示如方程式(2.30)、方程式(2.31)與方程式(2.32)，其中 c、e 爲分別非等式拘束條件的數目與等式拘束條件的數目。方程式(2.30)與方程式(2.31)皆將違反量自乘一次，得其平方項，使其必大於零。因此原本的最佳化問題改寫成了沒有拘束條件的最佳化問題。但因“懲罰函數”與原“目標函數”比例(scale)上須調配恰當，也就是調整 $\mathbf{r}$，許多文獻採用逐漸放大懲罰項的概念將設計點循序導入可行解空間，然而此一作法會使整體最佳化目標 T 具有數值上的不良條件(ill-condition)，亦即當演算法逐漸收斂，T 也變得越來越難尋找最佳解。如何加入“適當”的懲罰項是個文獻中廣泛被討論且棘手的難題，見文獻 [1]。

$$\min T(\mathbf{d}, \mathbf{r}) = f(\mathbf{d}) + F(\mathbf{g}(\mathbf{d}), \mathbf{h}(\mathbf{d}), \mathbf{r}) \tag{2.29}$$

$$F_g(\mathbf{d}) = \sum_{j=1}^{c} \left[\max\{0, g_i(\mathbf{d})\}\right]^2 \tag{2.30}$$

$$F_h(\mathbf{d}) = \sum_{j=1}^{e} \left[h_j(\mathbf{d})\right]^2 \tag{2.31}$$

$$F(\mathbf{g}(\mathbf{d}), \mathbf{h}(\mathbf{d}), \mathbf{r}) = F(F_g(\mathbf{d}), F_h(\mathbf{d}), \mathbf{r}) \tag{2.32}$$

爲了處理可靠度最佳化的問題，需同時考慮可靠度與最佳化演算法，在了解可靠度方法的發展與優劣後，爲了改善以往可靠度精度不足的問題，使用以樣本法爲基礎的可靠度計算方法，並降低樣本法的樣本數來增進其效率與可行性。最佳化演算法使用以梯度法爲基礎的 SQP 演算法，除了計算出穩健的機率梯度外，並希望在最佳化中使用有別於懲罰函數的方法，來避免不良條件的發生。整體而言，本文所提出的演算法具有以下特性:

- 使用樣本法爲基礎來估計可靠度，並降低取樣的數量
- 融合 SQP 與樣本法來解決可靠度最佳化的問題
- 找出穩健的計算機率梯度的方法以供 SQP 使用
- 找尋替代懲罰函數與障礙函數的新作法

第 三 章　取樣 SQP 演算法

3.1　演算法大綱

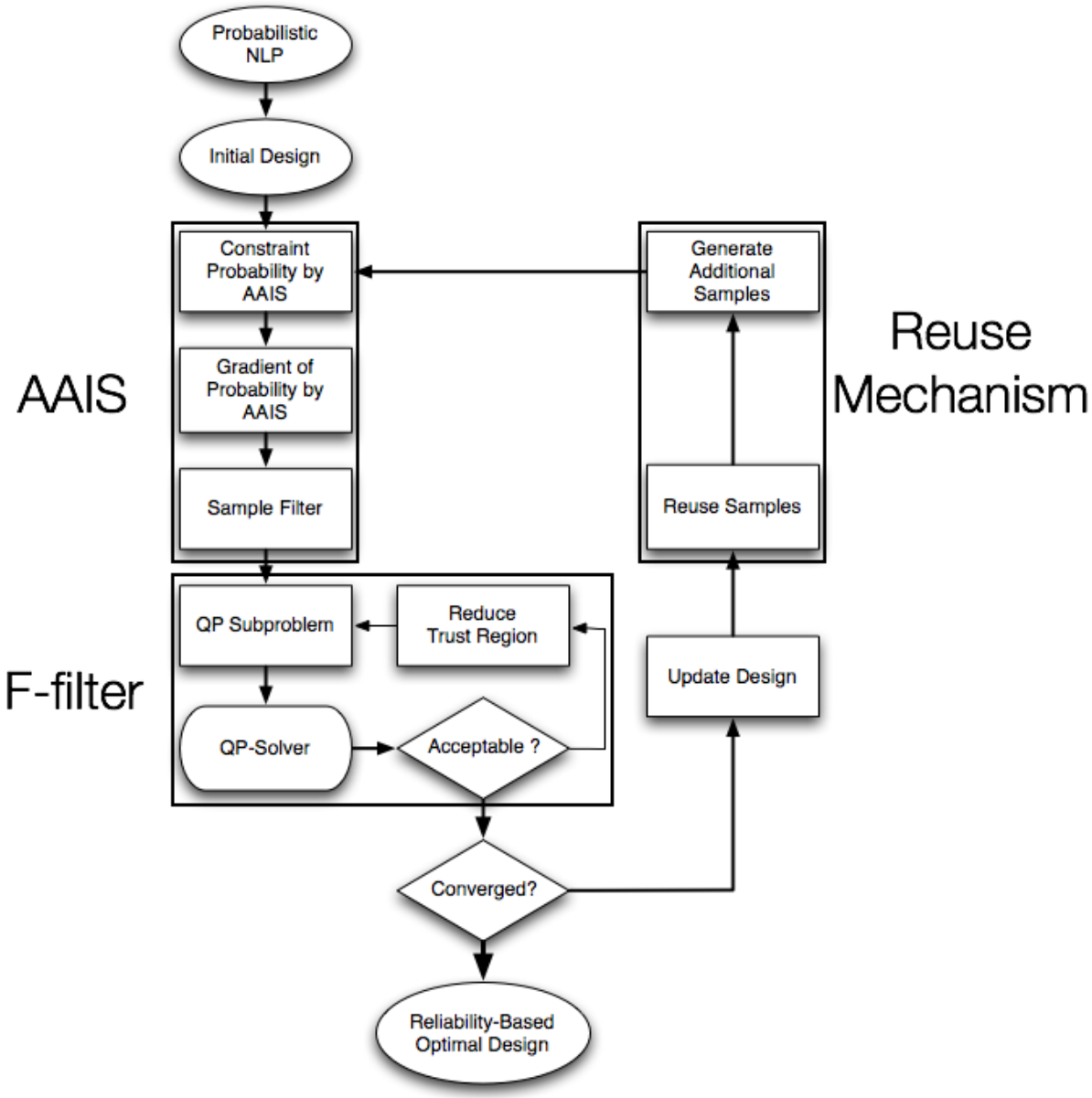

圖 3.1: 取樣 SQP 演算法流程圖

圖3.1爲本文提出的演算法流程圖，主要分成三部分來說明此次提出的演算法，AAIS取樣法、設計點F-filter試驗準則、樣本回收機制(reuse mechanism)。

首先定義問題並給予其初始點(initial point) $\boldsymbol{\mu}_{\mathbf{X}}^{0}$ 與初始的信賴區間(trust region) $\boldsymbol{\rho}^{0}$ ，上標 k 代表第 k 次疊代過程，再利用 AAIS(approach of average importance sampling)來估計破壞機率($\hat{P}_{\mathrm{f},j}^{k}$)與其破壞機率梯度($\nabla\hat{P}_{\mathrm{f},j}^{k}$)，其方法爲利用樣本所獲得的結果，並平均來獲得所需的資訊。但因AAIS是樣本法的一種，樣本法在計算時可能有偏差(bias)的產生，爲了避免有偏差的破壞機率梯度資訊被計算在內，而影響平均的破壞機率梯度，因而設置了樣本過濾器(sample filter)，來防止有異常值(outlier)被加入計算，而影響破壞機率梯度的結果。

利用破壞機率與破壞機率梯度，可將考慮不確定因素的拘束條件，做泰勒一階展開，並同時對目標函數做泰勒二階展開，形成二次規劃的問題。因在最佳化的過程中需要一些疊代過程(iteration)來找尋最佳解(mininum)，而每次的疊代過程中皆會有一個二次規劃問題待解決(因爲 $\boldsymbol{\mu}_{\mathbf{X}}^{k}$的不同)，致此將每次疊代過程中的二次規劃問題，稱爲二次規劃的子問題(QP subproblem)，子問題如方程式(3.1)所示，其中上標 k 代表第 k 次疊代過程， ∇f^{k} 與 $\mathbf{H}^{k}$ 分別爲目標函數的梯度與 Hessian 矩陣。

$$\mathrm{QP}^{k}(\boldsymbol{\mu}_{\mathbf{X}}^{k},\rho^{k})=\begin{cases}\min\limits_{\mathbf{s}} & \dfrac{1}{2}\mathbf{s}^{T}\mathbf{H}^{k}\mathbf{s}+(\nabla f^{k})^{T}\cdot\mathbf{s}\\ \text{s.t.} & \hat{P}_{j}^{k}+(\nabla\hat{P}_{j}^{k})^{T}\cdot\mathbf{s}\leq P_{\mathrm{f}}^{\mathrm{allow}}\\ & \|\mathbf{s}\|\leq\rho^{k}\end{cases}\tag{3.1}$$

在此進入了 F-filter 演算法的內容，在一般解決二次規劃的最佳化問題後，可得一步伐向量 $\mathbf{s}$(step vector)，來得到試驗點(trial point) $\boldsymbol{\mu}_{\mathbf{X}}^{k}+\mathbf{s}$ ，經由試驗準則來判斷試驗點是否通過。若通過則 $(\boldsymbol{\mu}_{\mathbf{X}}^{k}+\mathbf{s})$ 爲 $k+1$ 次疊代過程的起始點 $\boldsymbol{\mu}_{\mathbf{X}}^{k+1}$ ，若不通過，重新再解原二次規劃的子問題，但信賴區間 $\boldsymbol{\rho}^{k}$ (trust region)須減半，直到有試驗點通過試驗準則，產生下一次疊代的起始點爲止。因爲信賴區間 $\boldsymbol{\rho}^{k}$ 減半，所解出的結果會與原問題無減半的不同，並且在下一次的疊代過程當中，信賴區間會被更新，即是延用上一次疊代過程中的信賴區間，在此試驗準則會在3.5節有更詳細的介紹。

在這所提出的演算法，爲了降低計算成本，會回收之前疊代過程中所用過部份或全部的樣本資訊，來達到簡省成本的目的，即樣本回收機制(reuse mechanism)。之後更新設計變數，而整個演算法會持續疊代過程，直到滿足 KKT(Karush-Kuhn-Tucker conditions) 收斂條件，可參考文獻 [1]。下一小節會介紹 AAIS 及其如何使被用來估計考慮不確定因素下拘束條件的破壞機率，及其破壞機率梯度。

3.2 AAIS 取樣法

AAIS 的概念與2.2.7小節中的條件期望值(CE)非常相似，皆需選定一個隨機變數，來估計可靠度，但在Ayyub等人所提出的條件期望值並未對選定的變數(控制變數)做詳細的定義，見文獻 [24]。而 Royset 等人提出的 AAIS 有對選定的變數作較嚴格的限制，可參考文獻 [8]。在本文論述 AAIS 時以“控制變數”來稱呼其選定的變數，雖然在 Royset 等人的文獻中並未有控制變數的說法，但因與 CE 中的概念十分雷同，因此本文延用此說法來稱此被選定的隨機變數。

AAIS 以數學的角度爲出發點，經某一些假設，將方程式(1.6)推導出與條件期望值類似的計算破壞機率方式，並將其結果微分，來獲得破壞機率梯度，具有完整的數學背景與理論基礎。然而 AAIS 在 Royset 等人推導當中，選擇的控制變數爲一個隨機的參數(例如已知平均值與標準差的高斯分佈) [8]，但在本文中，將推導可選取的控制變數範圍，延伸至考慮不確定因素的設計變數(random variable)，並廣展隨機變數間的關係。Royset 等人，是假設全部隨機變數皆爲獨立，而本文則是推廣至控制變數與其餘隨機變數間爲獨立即可，此爲與 Royset 等人的 AAIS 不同之處。

3.2.1 AAIS 取樣法之假設

AAIS 是一種可以使用較少的樣本且可獲得精確破壞機率的一種方法，但要使用AAIS 來計算拘束條件的可靠度的拘束條件，需滿足下列幾項假設:

- 拘束條件函數的極限狀態(limit-state)爲連續一階可微($\mathcal{C}^1$)
  - 函數(Function):
    非空集合 A 中所有元素，恰有集合B的一元素相對應，此對應的關係稱爲由 A 映至 B ，一般將 A 稱爲定義域 B 爲值域或對應域，然而函數f可表示成:
    $f : A \rightarrow B \quad \text{or} f(A) = B$
  - 函數連續(Continuous):
    函數 f 在某個點 c 處是連續的當以下的兩個條件滿足：
    * $f(c)$ 必須是可定義的(即 c 必須是函數 f 的定義域中的元素)

* 如果 c 是定義域中的一點，則當 x 接近 c 時， $f(x)$ 的極限存在且等於 $f(c)$

– 函數一階可微(Differentiable):
  在函數的定義域中，函數的一階導數(derivative)存在

- 在拘束條件中至少有一個隨機變數的反函數存在，稱滿足此假設的變數爲控制變數(X_1 或經標準化後的 U_1)，並且選定的控制變數與其餘隨機變數間互爲獨立

  – 反函數(Inverse Function):
    $f : A \to B$ 爲一對一且映成函數，則具有反函數 $f^{-1} : B \to A$ 。

    * 一對一函數(one-one functin):若一函數將定義域中相異的元素對應到對映域中相異的元素，則稱此函數爲一對一函數，或稱嵌射(injection function)。

    * 映成函數(surjection function):若對映域中的每一個元素均可在定義域中找到一元素與之相對應，則稱此函數爲映成函數(onto function)，或稱蓋射，也有人稱爲滿射。

以下的推導，所選定的控制變數爲設計變數內的隨機變數，或說是帶有設計性質的隨機變數。

3.2.2 破壞機率與破壞機率梯度估計的推導

方程式(3.2)爲可靠度最佳化問題中常見的數學模型，符號定義同方程式(1.5)，並假設隨機變數皆爲高斯分布，在處理可靠度最佳化問題時最大的困難之一，就是如何計算拘束條件的破壞機率。一般在考慮某一拘束條件 $g_j(\mathbf{d}, \mathbf{U})$ 並對其進行破壞機率估計可寫成方程式(3.3)。在此 I_g 爲指標函數， ϕ_{n_r} 隨機變數的聯合機率密度函數， N 爲全部試驗的樣本數，當 $N \to \infty$ 時利用樣本法所估計的破壞機率 $(\hat{P}_{\mathrm{f},j})$ 會近似眞實的破壞機率 $(P_{\mathrm{f},j})$ 。

$$
\begin{gathered}
\min_{\mathbf{d}=[\mathbf{x_d}, \boldsymbol{\mu}_\mathbf{X}]} f(\mathbf{d}) \\
\text{subject to} \;\; P_{\mathrm{f},j} = \Pr[g_j(\mathbf{d}, \mathbf{U}) > 0] \leq P_{\mathrm{f},j}^{\mathrm{allow}} \;\; j = 1, \cdots, m \\
\mathbf{d}^{\mathrm{LB}} \leq \mathbf{d} \leq \mathbf{d}^{\mathrm{UB}} \\
\forall \mathbf{x_d} \in \Re^{n_d}, \; \boldsymbol{\mu}_\mathbf{X} \in \Re^{n_r}
\end{gathered}
\tag{3.2}
$$

$$
\begin{aligned}
P_{\mathrm{f},j} = \Pr[g_j(\mathbf{d},\mathbf{U}) > 0] &= \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} I_g(\mathbf{d},\mathbf{u}) \phi_{n_r}(\mathbf{u}) du_1 \cdots du_{n_r} \\
&\approx \frac{1}{N} \sum_{i=1}^{N} I_g(\mathbf{d},\mathbf{u}) \qquad (3.3)
\end{aligned}
$$

$$
I_g(\mathbf{d},\mathbf{u}) = \begin{cases} 1, \text{if } g_j(\mathbf{d},\mathbf{u}) > 0 \\ 0, \text{if } g_j(\mathbf{d},\mathbf{u}) \leq 0 \end{cases} .
$$

使用AAIS 的方法很簡單，首先產生 $n_r - 1$ 維的隨機變數樣本， $n_r - 1$ 維度的選取爲全部的隨機變數，除了控制變數 U_1 以外的其餘隨機變數，也可說就是除了滿足3.2.1節中可找到反函數的一個隨機變數之外的 $n_r - 1$ 維度的隨機變數。若滿足可找到反函數的隨機變數不只一維，由2.2.7小節，提到的文獻 [5]，建議選擇變異較大的隨機變數作爲控制變數。再產生 $n_r - 1$ 維的隨機變數樣本 N 組，接著再計算出可令拘束條件在極限狀態($g_j(\mathbf{d},\mathbf{u}) = 0$)下的 $u_{1,r}$ 值，其中 $r = 1, 2, 3 \cdots, N$，N 爲總樣本數。此外已知 U_1 隨機變數分佈，輕易的可計算出其 u_1 在累積分佈函數內的值，最後加總平均來獲得拘束條件的破壞機率。再次強調拘束條件需滿足3.2.1節中的假設，至少某一隨機變數的反函數存在且拘束條件在極限狀態下連續可微，滿足以上條件才可使用AAIS。

接著介紹方程式推導過程，在此先將控制變數在符號上獨立出來，因此 $\mathbf{u_1}$ 可用 $\bar{\mathbf{u}}$ 及 $\mathbf{d}$ 來表示，因爲該控制變數反函數存在。因此在極限狀態下，即 $g_j(\mathbf{d},\mathbf{u}) = 0$ 時，可表示如方程式(3.4)，其中 $\bar{\mathbf{u}} = [u_2, u_3, \cdots, u_{n_r}]$ 以及 $\mathbf{u} = [u_1, \bar{\mathbf{u}}]$

$$
\mathbf{u_1} = h_j(\mathbf{d}, \bar{\mathbf{u}}) \qquad (3.4)
$$

h_j 爲相對應的第 j 個拘束條件下，某特定選定的隨機變數的反函數，也就是控制變數的反函數。依拘束條件的型態不同， h_j 所對應的型態也不同，分爲下列兩種，小於型態、大於型態，如方程式(3.5)，其中 $\mathbf{u} = [\mathbf{u_1}, \bar{\mathbf{u}}]$，當 h_j 爲小於型態時，拘束條件的破壞機率可寫成如方程式(3.6)，其中 Φ 與 ϕ 分別爲高斯累積分佈函數(CDF)與高斯機率密度函數(PDF)，推導過程如方程式(3.6)。因控制變數與其餘隨機變數間的關係爲獨立，所以控制變數的機率密度函數可從聯合機率密度函數中獨立出來，其中 ϕ_{n_r} 爲其餘隨機變數的聯合機率密度函數，再利用獲得的反函數作爲積分的上下限，來加以積分。

$$
\begin{aligned}
&\text{當}\mathbf{u_1} \le h_j(\mathbf{d},\bar{\mathbf{u}})\text{滿足} \quad g_j(\mathbf{d},\mathbf{u}) > 0 \quad \text{則稱爲小於型態 (n-type)} \\
&\text{當}\mathbf{u_1} \ge h_j(\mathbf{d},\bar{\mathbf{u}})\text{滿足} \quad g_j(\mathbf{d},\mathbf{u}) > 0 \quad \text{則稱爲大於型態 (p-type)}
\end{aligned} \tag{3.5}
$$

$$
\begin{aligned}
P_{\mathrm{f},j} &= \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} I_g(\mathbf{d},\mathbf{u})\phi_{n_r}(\mathbf{u}) du_1 \ldots du_{n_r} \\
&= \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \left(\int_{-\infty}^{h_j(\mathbf{d},\bar{\mathbf{u}})} \phi_{u_1} du_1 \right) \phi_{n_r-1}(\bar{\mathbf{u}}) du_2 \ldots du_{n_r} \\
&= \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \Phi(h_j(\mathbf{d},\bar{\mathbf{u}}))\phi_{n_r-1}(\bar{\mathbf{u}}) du_2 \ldots du_{n_r}.
\end{aligned} \tag{3.6}
$$

又拘束條件對設計變數 $\mathbf{d}$ 滿足連續一階可微，因此破壞機率的梯度，可藉由計算破壞機率方程式(3.6)，微分來獲得計算破壞機率梯度的方程式，過程如方程式(3.7)所示，觀察方程式(3.7)可知需推導 $\nabla_{\mathbf{d}} h_j(\mathbf{d},\bar{\mathbf{u}})$ ，又 $g_j(\mathbf{d},(h_j(\mathbf{d},\bar{\mathbf{u}}),\bar{\mathbf{u}}))$ 會滿足極限狀態，即 $g_j(\mathbf{d},(h_j(\mathbf{d},\bar{\mathbf{u}}),\bar{\mathbf{u}})) = 0$，因此 $\nabla_{\mathbf{d}} g_j(\mathbf{d},(h_j(\mathbf{d},\bar{\mathbf{u}}),\bar{\mathbf{u}})) = 0$。經過連鎖率(chain rule)可得到 $\nabla_{\mathbf{d}} h_j(\mathbf{d},\bar{\mathbf{u}})$，如方程式(3.8)，在此須注意方程式3.8中的分母不可爲 0，理論上 $\partial g_j(\mathbf{d},(h_j(\mathbf{d},\bar{\mathbf{U}}),\bar{\mathbf{U}}))/\partial u_1$ 爲一個不會爲0的量值，若爲 0 則隱含拘束條件違反3.2.1節中可微的條件。此外還須注意，當 $\partial g_j(\mathbf{d},(h_j(\mathbf{d},\bar{\mathbf{U}}),\bar{\mathbf{U}}))/\partial u_1 \approx 0$ 造成計算梯度時有偏差出現，爲了防止這種現象，因而在演算中增設樣本過濾器，在3.3小節中會有更詳細的介紹。將方程式(3.8)代入方程式(3.7)可得估計破壞機率梯度的方程式(3.9)

$$
\begin{aligned}
\nabla_{\mathbf{d}} P_{\mathrm{f},j}(\mathbf{d}) &= \nabla_{\mathbf{d}} \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \Phi(h_j(\mathbf{d},\bar{\mathbf{u}}))\phi_{n_r-1}(\bar{\mathbf{u}}) du_2 \ldots du_{n_r} \\
&= \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \nabla_{\mathbf{d}} \Phi(h_j(\mathbf{d},\bar{\mathbf{u}}))\phi_{n_r-1}(\bar{\mathbf{u}}) du_2 \ldots du_{n_r} \\
&= \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \phi(h_j(\mathbf{d},\bar{\mathbf{u}}))\nabla_{\mathbf{d}} h_j(\mathbf{d},\bar{\mathbf{u}})\phi_{n_r-1}(\bar{\mathbf{u}}) du_2 \ldots du_{n_r}
\end{aligned} \tag{3.7}
$$

又

$$
\begin{aligned}
&\nabla_{\mathbf{d}} g_j + \frac{\partial g_j(\mathbf{d},(h_j(\mathbf{d},\bar{\mathbf{u}}),\bar{\mathbf{u}}))}{\partial h_j(\mathbf{d},\bar{\mathbf{u}})} \nabla_{\mathbf{d}} h_j \\
&= \nabla_{\mathbf{d}} g_j + \frac{\partial g_j(\mathbf{d},(h_j(\mathbf{d},\bar{\mathbf{u}}),\bar{\mathbf{u}}))}{\partial u_1} \nabla_{\mathbf{d}} h_j
\end{aligned}
$$

所以

$$\nabla_{d_i} h_j(\mathbf{d}, \bar{\mathbf{U}}) = -\frac{\nabla_{d_i} g_j(\mathbf{d}, (h_j(\mathbf{d}, \bar{\mathbf{U}}), \bar{\mathbf{U}}))}{\partial g_j(\mathbf{d}, (h_j(\mathbf{d}, \bar{\mathbf{U}}), \bar{\mathbf{U}}))/\partial u_1} \tag{3.8}$$

$$\nabla_{\mathbf{d}} P_{\mathrm{f},j}(\mathbf{d}) = -\int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \Phi(h_j(\mathbf{d}, \bar{\mathbf{u}})) \frac{\nabla_{\mathbf{d}} g_j(\mathbf{d}, (h_j(\mathbf{d}, \bar{\mathbf{u}}), \bar{\mathbf{u}}))}{\partial g_j(\mathbf{d}, (h_j(\mathbf{d}, \bar{\mathbf{u}}), \bar{\mathbf{u}}))/\partial u_1} \phi_{n_r-1}(\bar{\mathbf{u}}) d\bar{\mathbf{u}} \tag{3.9}$$

當 N 組樣本產生後，方程式(3.6)、方程式(3.7)，可改寫成方程式(3.10)與方程式(3.11)，轉換原理如同方程式(2.19)等效於方程式(2.18)。

$$\hat{P}_{\mathrm{f},j}(\mathbf{d}) \approx \frac{1}{N} \sum_{r=1}^{N} \Phi(h_j(\mathbf{d}, \bar{\mathbf{u}}_r)) \tag{3.10}$$

$$\nabla_{\mathbf{d}} \hat{P}_{\mathrm{f},j}(\mathbf{d}) \approx \frac{1}{N} \sum_{r=1}^{N} \phi(h_j(\mathbf{d}, \bar{\mathbf{u}}_r)) \nabla_{\mathbf{d}} h_j \tag{3.11}$$

當 h_j 爲大於型態時，處理方法相同，拘束條件的破壞機率可寫成如方程式(3.12)。

$$\begin{aligned} P_{\mathrm{f},j} &= \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} I_g(\mathbf{d}, \mathbf{u}) \phi_{n_r}(\mathbf{u}) du_1 \ldots du_{n_r} \\ &= \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \left(\int_{h_j(\mathbf{d}, \bar{\mathbf{u}})}^{\infty} \phi_{u_1} du_1 \right) \phi_{n_r-1}(\bar{\mathbf{u}}) du_2 \ldots du_{n_r} \\ &= \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \Phi(-h_j(\mathbf{d}, \bar{\mathbf{u}})) \phi_{n_r-1}(\bar{\mathbf{u}}) du_2 \ldots du_{n_r}. \end{aligned} \tag{3.12}$$

如同方程式(3.7)，藉由估計破壞機率方程式(3.12)，微分來獲得計算破壞機率梯度的方程式，過程如方程式(3.13)所示:

$$\begin{aligned} \nabla_{\mathbf{d}} P_{\mathrm{f},j}(\mathbf{d}) &= \nabla_{\mathbf{d}} \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \Phi(-h_j(\mathbf{d}, \bar{\mathbf{u}})) \phi_{n_r-1}(\bar{\mathbf{u}}) du_2 \ldots du_{n_r} \\ &= \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \nabla_{\mathbf{d}} \Phi(-h_j(\mathbf{d}, \bar{\mathbf{u}})) \phi_{n_r-1}(\bar{\mathbf{u}}) du_2 \ldots du_{n_r} \\ &= \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \phi(-h_j(\mathbf{d}, \bar{\mathbf{u}})) \nabla_{\mathbf{d}} h_j(\mathbf{d}, \bar{\mathbf{u}}) \phi_{n_r-1}(\bar{\mathbf{u}}) du_2 \ldots du_{n_r} \end{aligned} \tag{3.13}$$

將方程式(3.8)代入方程式(3.13)可得估計破壞機率梯度的方程式(3.14)

$$\nabla_{\mathbf{d}} P_{\mathrm{f},j}(\mathbf{d}) = -\int_{-\infty}^{\infty}\cdots\int_{-\infty}^{\infty}\Phi(-h_j(\mathbf{d},\bar{\mathbf{u}}))\frac{\nabla_{\mathbf{d}} g_j(\mathbf{d},(h_j(\mathbf{d},\bar{\mathbf{u}}),\bar{\mathbf{u}}))}{\partial g_j(\mathbf{d},(h_j(\mathbf{d},\bar{\mathbf{u}}),\bar{\mathbf{u}}))/\partial u_1}\phi_{n_r-1}(\bar{\mathbf{u}})d\bar{\mathbf{u}} \tag{3.14}$$

當 N 組樣本產生後，方程式(3.12)、方程式(3.13)，可改寫成方程式(3.15)與方程式(3.16)，轉換原理如同方程式(2.19)等效於方程式(2.18)。

$$\hat{P}_{\mathrm{f},j}(\mathbf{d}) \approx \frac{1}{N}\sum_{r=1}^{N}\Phi(-h_j(\mathbf{d},\bar{\mathbf{u}}_r)) \tag{3.15}$$

$$\nabla_{\mathbf{d}}\hat{P}_{\mathrm{f},j}(\mathbf{d}) \approx \frac{1}{N}\sum_{r=1}^{N}\phi(-h_j(\mathbf{d},\bar{\mathbf{u}}_r))\nabla_{\mathbf{d}} h_j \tag{3.16}$$

3.2.3 AAIS 取樣法之討論

由上一小節已知 AAIS 的數學推導過程，本小節則探討其推導出的計算方法所代表的意義，以 n-type 情形所推導出來的計算方程式(3.10)與方程式(3.11)爲例。首先討論機率的計算公式3.10，若將其分解來看，其中 $h_j(\mathbf{d},\bar{\mathbf{u}}_r)$ 的意義如同於圖3.2，取得其除了控制變數以外的樣本 $\bar{\mathbf{u}}$ ，再利用控制變數的反函數，求得拘束條件在極限狀態下，控制變數所對應的值 $u_{1,r}$ $(u_{1,r} = h_j(\mathbf{d},\bar{\mathbf{u}}_r))$，也就是 $h_j(\mathbf{d},\bar{\mathbf{u}}_r)$使用的意義找出相對應的 u_1 值，也就是其爲反函數的意義。

控制變數所對應的數值處於一個臨界位置，破壞與安全的交界(因爲滿足極限狀態)，如圖3.2所示。因此若知道控制變數的機率密度函數 f_{U_1} 後，可視爲 f_{U_1} 被 $u_{1,r}$ 劃分爲兩個區域，一邊是破壞區域，一邊是安全區域，參考圖3.3，最後可利用其累積分佈函數 F_{U_1} 來計算其破壞機率，在此因爲假想爲高斯分佈，所以使用 $\Phi(h_j)$ 來估計，非高斯分佈的計算方法參考3.2.4小節。在反覆取樣 N 次後獲得 $u_{1,r}$，與估計的破壞機率 $\hat{P}_{\mathrm{f},j,r}$，在此 $r = [1, 2, \cdots, N]$，最後加總平均得到估計的破壞機率$\hat{P}_{\mathrm{f},j}$。因爲取樣在拘束條件的邊界上，並利用累積分佈函數來估算破壞機率，因此可知只要使用少許座落在邊界的樣本，就可獲得精確的估計。

而在破壞機率梯度方面，可分成梯度項 $\nabla_{d_i} g_j(\mathbf{d},(h_j(\mathbf{d},\bar{\mathbf{U}}),\bar{\mathbf{U}}))$ 與機率項 $\phi(h_j(\mathbf{d},\bar{\mathbf{u}}_r))$ 、

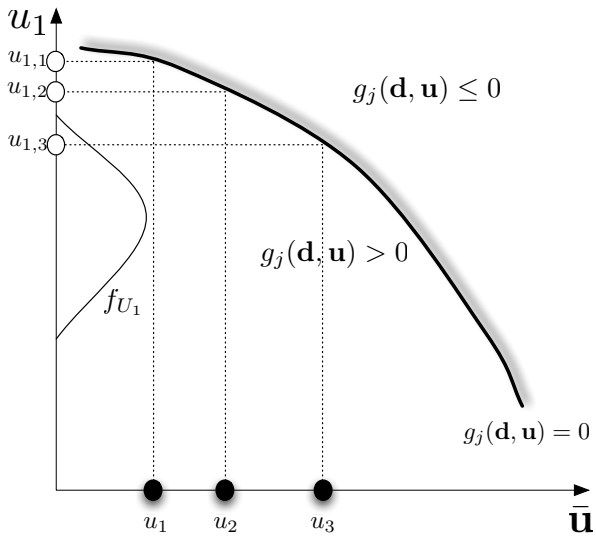

圖 3.2: AAIS 取樣法示意圖

$\partial g_j(\mathbf{d},(h_j(\mathbf{d},\bar{\mathbf{U}}),\bar{\mathbf{U}}))/\partial u_1$ ，因爲 $\phi((h_j(\mathbf{d},\bar{\mathbf{u}}_r))$ 的值所代表的是機率密度，而機率密度本身並無法代表機率，因此將兩項機率項合併的結果: $\frac{\phi(h_j(\mathbf{d},\bar{\mathbf{u}}_r))}{\partial g_j(\mathbf{d},(h_j(\mathbf{d},\bar{\mathbf{U}}),\bar{\mathbf{U}}))/\partial u_1}$ ，視爲將梯度調整爲機率梯度的比例項次。而梯度項則是在拘束條件的邊界上，即極限狀態的情形上求取梯度，因爲將反函數導入在拘束條件內，所以會是在極限狀態下求取梯度。使用較少的樣本數可計算出精確的破壞機率，同理，使用較少的樣本數可獲得精確的破壞機率梯度。

AAIS 會隨著樣本的增加，計算也越趨精準，而最大的優勢爲其利用少許的樣本就可計算出精準的結果。總結實行 AAIS 的步驟，可分爲下列幾個步驟:

- 確認與選擇控制變數
- 找出其反函數
- 產生隨機變數，除了控制變數外 $\bar{\mathbf{u}}$
- 計算 N 個對應的控制變數 $\mathbf{u_1}$
- 利用方程式(3.10)與(3.11)得到 $\hat{P}_{\mathrm{f},j}$ 與 $\nabla\hat{P}_{\mathrm{f},j}$

在此假想若反函數不唯一時會是一個如何的情形，或說是當反函數存在的假設不成立時，會是何種情形。此種情形就好比爲圖3.2中的任一組 $\bar{\mathbf{u}}_r$ ，對映的 $\mathbf{u}_{1,r}$ 不只一組的，然而，此種情形若反應在控制變數的機率密度函數時，有如想像圖3.3被許多的 $u_{1,r}$ 所切割，

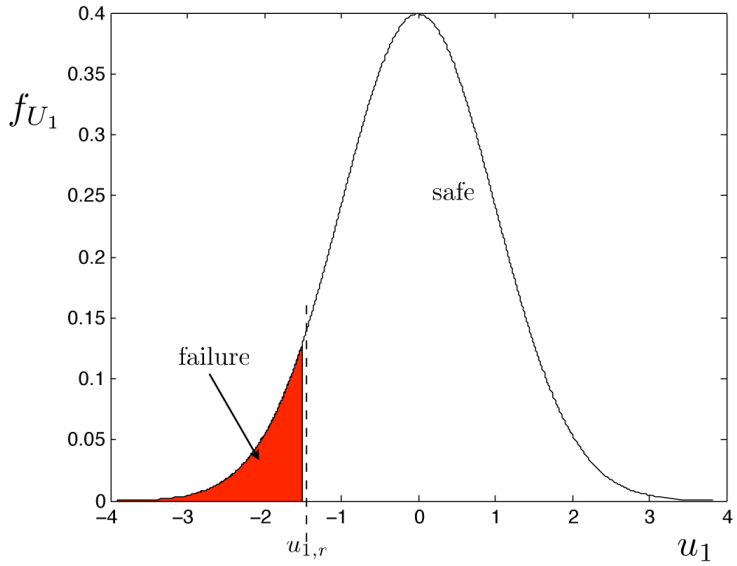

圖 3.3: 控制變數的機率密度函數

分成許多安全或破壞的區間。因此若要計算該破壞機率與其梯度，首要的任務就是要將安全與破壞的區間區隔出來。然而要將安全與破壞的區間辨識出來，必須先找出在各組 $\bar{\mathbf{u}}_r$ 所對應的控制變數的根(root)，若是產生虛根可以視爲無破壞情形發生，但往往尋根的過程是充滿困難與荊棘。

3.2.4 延伸至非高斯隨機變數

在方程式(3.10)、(3.11)與方程式(3.15)、(3.16)說明了在高斯分佈下如何用 AAIS 來計算破壞機率與其梯度。在工程領域中有些不確定因素不一定使用高斯隨機變數來模擬，如對數分佈等等⋯，因此 AAIS 因應此情況來做些許的修正。而方程式(3.10)、(3.11)可改寫成方程式(3.17)與方程式(3.18)，其中 F_{U_1} 與 f_{U_1} 分別爲該控制變數的累積分佈函數與機率密度函數。而方程式(3.15)、(3.16)可改寫成方程式(3.19)與方程式(3.20)

$$\hat{P}_{\mathrm{f},j}(\mathbf{d}) \approx \frac{1}{N}\sum_{r=1}^{N} F_{U_1}(h_j(\mathbf{d},\bar{\mathbf{U}}_r)) \tag{3.17}$$

$$\nabla_{\mathbf{d}}\hat{P}_{\mathrm{f},j}(\mathbf{d}) \approx \frac{1}{N}\sum_{r=1}^{N} f_{U_1}(h_j(\mathbf{d},\bar{\mathbf{U}}_r))\nabla_{\mathbf{d}} h_j \tag{3.18}$$

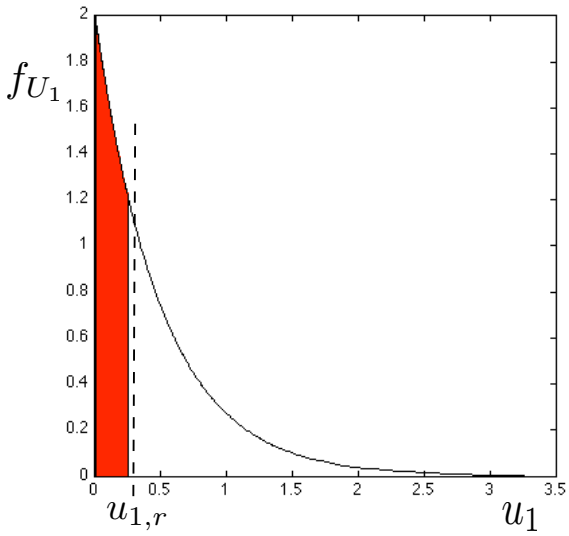

圖 3.4: 非高斯控制變數的機率密度函數

$$\hat{P}_{\mathrm{f},j}(\mathbf{d}) \approx \frac{1}{N}\sum_{r=1}^{N} F_{U_1}(-h_j(\mathbf{d},\bar{\mathbf{U}}_r)) \tag{3.19}$$

$$\nabla_{\mathbf{d}}\hat{P}_{\mathrm{f},j}(\mathbf{d}) \approx \frac{1}{N}\sum_{r=1}^{N} f_{U_1}(-h_j(\mathbf{d},\bar{\mathbf{U}}_r))\nabla_{\mathbf{d}}h_j \tag{3.20}$$

因此方程式(3.17)、(3.18)與方程式(3.19)、(3.20)可以處理帶有非高斯隨機變數的不確定因素的問題。在此可以將問題做簡單的分類，依控制變數的隨機分佈型式，分爲高斯與非高斯隨機變數。

- 若控制變數爲高斯分佈
  使用方程式(3.10)、(3.11)與方程式(3.15)、(3.16)來處理。
- 若控制變數爲非高斯分佈
  使用方程式(3.17)、(3.18)與方程式(3.19)、(3.20)來處理。

在此其餘維度的隨機變數，並不影響AAIS使用模式，因爲其餘維度的隨機變數只需依其機率密度來產生樣本 $\bar{\mathbf{u}}$ 即可。然而方程式轉換的道理很簡單，當控制變數爲高斯分佈時，利用控制變數的機率密度函數 f_{U_1} 以及 $u_{1,r}$ 的概念來計算破壞機率，如圖3.3。因此同理，當控制變數爲非高斯時，利用控制變數的機率密度函數 f_{U_1} 以及 $u_{1,r}$的概念來計算破壞機率，如圖3.4，最後可利用其累積分佈函數來計算其破壞機率。

3.2.5 結合重要取點法

爲了增進使用樣本法的效率與降低樣本數，在此結合2.2.5中介紹的重要取點法。重要取點法有兩個重大特色:改變隨機變數的參數與轉換隨機變數的分佈。若欲利用第一個性質，需對問題特質有相當程度的了解，取決於工程師對問題的定義與認知，因此在此不使用第一個性質，在此使用第二個性質，轉換隨機變數的分佈， 將方程式(3.10)與方程式(3.11)改寫成方程式(3.21)與方程式(3.22)，其中 $f_{\bar{U}}$ 爲控制變數以外的隨機變數的機率密度函數(PDF)，而 $f'_{\bar{U}}$ 爲使用者自定的隨機變數的機率密度函數(PDF)。若使用者自定的隨機變數分佈皆爲均勻分佈時，方程式(3.21)與方程式(3.22)可改寫成方程式(3.23)與方程式(3.24)

$$\hat{P}_{\mathrm{f},j}(\mathbf{d}) \approx \frac{1}{N}\sum_{r=1}^{N}\Phi(h_j(\mathbf{d},\bar{\mathbf{u}}_r))\frac{f_{\bar{U}}(\bar{\mathbf{u}}_r)}{f'_{\bar{U}}(\bar{\mathbf{u}}_r)} \tag{3.21}$$

$$\nabla_{\mathbf{d}}\hat{P}_{\mathrm{f},j}(\mathbf{d}) \approx \frac{1}{N}\sum_{r=1}^{N}\phi(h_j(\mathbf{d},\bar{\mathbf{u}}_r))\nabla_{\mathbf{d}}h_j\frac{f_{\bar{U}}(\bar{\mathbf{u}}_r)}{f'_{\bar{U}}(\bar{\mathbf{u}}_r)} \tag{3.22}$$

$$\hat{P}_{\mathrm{f},j}(\mathbf{d}) \approx \frac{1}{N}\sum_{r=1}^{N}\Phi(h_j(\mathbf{d},\bar{\mathbf{u}}_r))\frac{f_{\bar{U}}(\bar{\mathbf{u}}_r)}{1/2\boldsymbol{\Delta}} \tag{3.23}$$

$$\nabla_{\mathbf{d}}\hat{P}_{\mathrm{f},j}(\mathbf{d}) \approx \frac{1}{N}\sum_{r=1}^{N}\phi(h_j(\mathbf{d},\bar{\mathbf{u}}_r))\nabla_{\mathbf{d}}h_j\frac{f_{\bar{U}}(\bar{\mathbf{u}}_r)}{1/2\boldsymbol{\Delta}}. \tag{3.24}$$

均勻分佈的範圍爲 $[\mathbf{d}-\boldsymbol{\Delta},\mathbf{d}+\boldsymbol{\Delta}]$ 來取代原本的高斯分佈 $[-\infty,\infty]$ ， $\boldsymbol{\Delta}$ 可能會造成些微的誤差，造成誤差的原因可能爲取樣的樣本超出均勻分佈的範圍所導致的，但隨著 $\boldsymbol{\Delta}$ 的增加，造成誤差會越趨近 0 。 $\boldsymbol{\Delta}$ 值與隨機變數的標準差有直接的關連，因爲標準差決定了隨機變數出現在均勻分佈範圍的機率，所以在此選擇 $\boldsymbol{\Delta}=4\boldsymbol{\sigma}_{\bar{\mathbf{U}}}$ ，所以此均勻分佈可含括機率 99.99% 的高斯分佈區域。若 P_{f} 是極爲小的值時， $\boldsymbol{\Delta}$ 被修正，一般來說當 P_{f} 愈來愈小，則 $\boldsymbol{\Delta}$ 須增加來做修正。

最後當型態不同時，同理方程式(3.15)、方程式(3.16)轉換分佈，改爲均勻分佈後可表示成方程式(3.25)與方程式(3.26)

$$\hat{P}_{\mathrm{f},j}(\mathbf{d}) \approx \frac{1}{N}\sum_{r=1}^{N}\Phi(-h_j(\mathbf{d},\bar{\mathbf{u}}_r))\frac{f_{\bar{U}}(\bar{\mathbf{u}}_r)}{1/2\boldsymbol{\Delta}} \tag{3.25}$$

$$\nabla_{\mathbf{d}}\hat{P}_{\mathrm{f},j}(\mathbf{d}) \approx \frac{1}{N}\sum_{r=1}^{N}\phi(-h_j(\mathbf{d},\bar{\mathbf{u}}_r))\nabla_{\mathbf{d}}h_j\frac{f_{\bar{U}}(\bar{\mathbf{u}}_r)}{1/2\boldsymbol{\Delta}}. \tag{3.26}$$

3.3 樣本過濾器

在3.2.2節中，討論到破壞機率梯度的計算，如方程式(3.11)所示。但在現實當中，取樣數目必有其上限，不可能取樣至到無窮多組。除此之外，因爲計算破壞機率梯度的方法是取多組樣本的結果並平均之，因此在計算的過程中可能會受到異常值(outlier)的干擾。此外爲了預防方程式(3.11)中的 $\partial g_j(\mathbf{d},(h_j(\mathbf{d},\bar{\mathbf{U}}),\bar{\mathbf{U}}))/\partial u_1 \approx 0$ 造成計算出的破壞機率有偏差的情型發生因而設置樣本過濾器來避免此情型發生。

樣本過濾器的過濾機置爲，首先計算全部樣本的梯度資訊， 每一維度再依其大小做排列，擷取排列後梯度資訊中的 $\frac{t}{2}\% \sim \left(100-\frac{t}{2}\right)\%$。$t$ 爲使用者所設定擷取梯度資訊範圍的參數，將其排列後的資訊略過頭尾各 $\frac{t}{2}\%$，t 不宜選取過大，只需略去少量的梯度資訊即可，防止少數的異常值影響梯度結果，在此建議 $t\% \leq 5\%$。

3.4 樣本回收機器

爲了增進演算法的效率，因此增設了樣本回收機器。概念相當的簡單，就是將之前使用過的樣本回收再使用。若考慮某兩次連續的疊代過程中的設計點 $\mathbf{d}^k$、$\mathbf{d}^{k+1}$， 若在兩設計點做蒙地卡羅模擬，且兩點相距不遠時，不難想像會有不少的點座落在重複的區域內，此爲回收概念的最初理念。

在本次所提出的演算法，將除了控制變數外的隨機變數($\bar{\mathbf{u}}$)的分佈皆利用重要取點的特均轉換爲均勻分佈，如圖3.5所示，Ω 爲均勻分佈的範圍，因此當隨機分佈轉換爲均勻分佈後， $\bar{\mathbf{u}}^k$ 會全部座落在 Ω^k 內，同理 $\bar{\mathbf{u}}^{k+1}$ 會全部座落在 Ω^{k+1} 內，而兩區域重疊的部分 $\Omega_{\mathrm{R}}^{k+1}$，正是樣本可重複利用的區域，換句話說，$\Omega^k$ 與 Ω^{k+1} 交集範圍內的樣本可回收再利用。而 $\bar{\mathbf{u}}^{k+1}$ 的樣本有些許是來自 $\bar{\mathbf{u}}^k$，在 $k+1$ 的疊代過程中實際取樣的區域只有在 Ω^{k+1} 與重複區 $\Omega_{\mathrm{R}}^{k+1}$ 以外的區域，也就是 Ω^{k+1} 與 $\Omega_{\mathrm{R}}^{k+1}$ 間的差集，因爲隨機變數的分佈改爲均勻分佈，因此若 $\Omega_{\mathrm{R}}^{k+1}$ 的區域愈大，則需取樣的點就愈少。隨著步伐大小 ($\mathbf{s}$) 的不同會造成回收機制效率的不同，若步伐較大可能只有較少的樣本被回收，有時甚至沒有任何樣本被回收，若是步伐較小時則會有較多的樣本被回收。此外類似 SQP 這類以梯度爲基礎的演算法，都會有一個特性，就是在演算法的過程當中找尋到最佳值附近時，會在最佳解附近震

盪(zigzag)，花費許多計算量，但若可以回收樣本後，可以將在最佳值附近震盪所帶來的計算量大大降低，有時有可能將上一次的樣本完全回收，因爲移動的步伐很小。使用回收機制的唯一限制就是全部的設計變數需爲隨機變數。

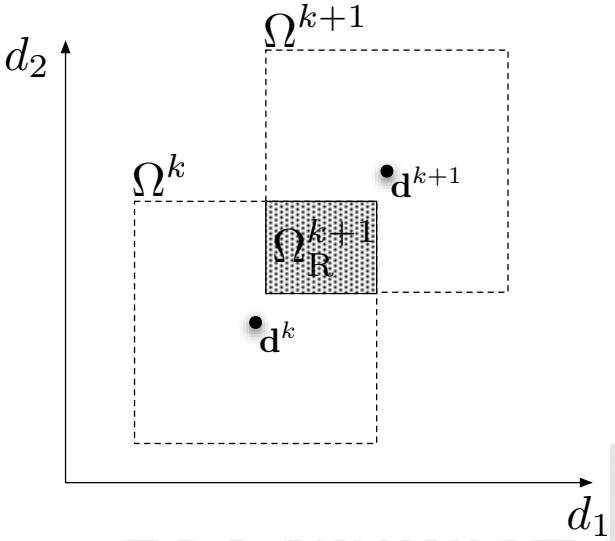

圖 3.5: 回收機制示意圖

將隨機變數的分佈轉爲均勻分佈的原因有二: 不容易維持分佈的特性、回收效率較差。若自定的分佈 $f_{\bar{U}'}$ 不使用均勻分佈，舉例改使用高斯分佈時，因爲 $\mathbf{d}^k$ 與 $\mathbf{d}^{k+1}$ 必不同，所以若將重複部分之樣本全部回收，會造成爲維持隨機分佈的性質而造成取樣數目的增加，但若又爲了維持隨機分佈不將全部重複部分樣本回收，則造成樣本法效率的低落。因此選均勻分佈做爲自選的隨機分佈 $f'_{\bar{U}'}$ 來兼顧效率與保持隨機分佈的特性。而回收機制的整體流程可分別下列四步:

- 利用 $\mathbf{\Delta}$ 與 $\mathbf{d}^k$、$\mathbf{d}^{k+1}$ 判斷回收的範圍
- 獲得回收的樣本
- 計算回收樣本數與需再取樣的數目
- 產生落在 Ω^{k+1} 與 $\Omega_{\mathrm{R}}^{k+1}$ 差集的新樣本

3.5 試驗機制

在本篇論文中，嘗試使用 Fletcher 的 F-filter，來取代懲罰函數 [39] 。 因此在本文當中所使用的演算法，並不會完全接受二次規劃子問題(subproblem)的結果，即不是每個二次規劃子問題的結果都會成爲下一疊代點。需端看二次規劃子問題的結果，是否可滿足某些特定的試驗準則，若滿足則成爲下一疊代點，但若不滿足，需重新再解一次原二次規劃子問題，且信賴區域(trust region)須減半，因此成爲一個判斷，試驗點是否被接受的過濾器演算法，在此稱爲 F-filter 演算法，而此演算法是由 Fletcher 等人所提出的 [39]。 F-filter 演算法提供了全局收斂証明，証明在恰當使用的前提下，此 F-filter 演算法可導致這一連串的二次規劃子問題最終可以收斂。

在介紹 F-filter 演算法前，先定義兩個名詞”不可行性”(infeasibility)與”支配”(dominance)。首先介紹不可行性，在第 k 次疊代中的 $\mathbf{d}^k$ 所對應的”不可行性” v^k 爲全部拘束條件中最大的違反量，在此上標 k 表示第 k 次疊代，數學形式可寫成方程式(3.27)

$$v^k(\mathbf{g}'(\mathbf{d}^k)) = \max_j(0, g'_j(\mathbf{d}^k)) \tag{3.27}$$

若拘束條件不考慮可靠度，則 $g'_j(\mathbf{d}^k) = g_j(\mathbf{d}^k)$ ，若拘束條件考慮可靠度時，則 $g'_j(\mathbf{d}^k) = \hat{P}^k_{\mathrm{f},j}(\mathbf{d}^k) - P^{\mathrm{allow}}_{j,\mathrm{f}}$ ，其中 $\hat{P}^k_{\mathrm{f},j}(\mathbf{d}^k)$ 爲第 j 個拘束條件被估計的破壞機率， $P^{\mathrm{allow}}_{j,\mathrm{f}}$ 爲第 j 個拘束條件所被允許的最大破壞機率。同時也可估計第 k 次疊代的目標函數 $f^k(\mathbf{d}^k)$ ，因此目標函數值與不可行性已知爲 $\{f^k(\boldsymbol{\mu}^k_{\mathbf{X}}), v^k(\mathbf{g}'(\boldsymbol{\mu}^k_{\mathbf{X}}))\}$ ，此稱爲第 k 次疊代的結果。進而可對每一次疊代結果做比較，然而”支配”正是一種比較結果後的現象，若在第 k 次疊代中，該次設計變數 $\mathbf{d}^k$ 所帶來的結果與之前結果相比較，發現若 $v^k \geq v^l$ 且 $f^k \geq f^l$ 的條件成立時，定義爲第 k 次疊代結果被第 l 次疊代結果所支配，在此可知當方程式(3.28)與方程式(3.29)同時成立時，稱爲”被支配”的條件。

$$v^k \geq v^l \tag{3.28}$$

$$f^k \geq f^l \tag{3.29}$$

利用圖3.6再次說明”支配”的意義，如圖3.6中所示 D 點被 B 點所支配，這就是一個最簡單的被支配的例子。然而在圖3.6中的實線是由某次疊代過程目前刻下過濾器內的結果所接連起來的，A、B、C 三點則是其中的代表。因此可知過濾器就是一個包含某一些疊代結

果$\{\mathbf{f},\mathbf{v}\}$的集合，並且過濾器中的點會隨疊代過程的進行可能有所增加，反之過濾器內不一定包含全部的疊代結果。 爲了提升收斂的特質，Fletcher 等人建議使用兩個非零的參數 α

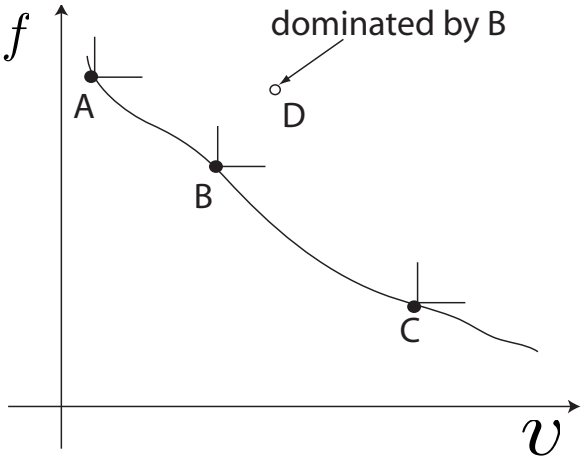

圖 3.6: F-filter 過濾器概念示意圖

與 γ ，來轉換”支配”的條件 [39]，在 Fletcher 的演算法中若該次疊代結果被過濾器內的點所支配，則稱爲不被過濾器所接受(acceptable)，反之亦然。因此”不被過濾器接受”的條件可寫成方程式(3.30)與方程式(3.31)，其中 $\{f,v\}$ 爲此次疊代的結果，且 $1>\alpha>\gamma>0$ ，其中 α 爲極接近 1 的值，而 γ 則是極爲接近0的值，因爲 $1-\alpha$ 與 γ 皆爲極小的值，因此可以忽略方程式(3.30)、方程式(3.31)與方程式(3.28)、方程式(3.29)之間的差異。而 $l\in\mathcal{F}$ ， $\mathcal{F}$ 爲該次過濾器內所包含的結果其第 k 次疊代的集合，例: $k=5$，$l\in\{2,5,7,10,\cdots\}$ ， $\mathcal{F}=\{2,5,7,10,\cdots\}$ 。若該次子問題的結果不被過濾器所接受時，則將信賴區域(trust region)減半，再解此二次規劃問題。

$$v \geq \alpha v^l \tag{3.30}$$

$$f \geq f^l-\gamma v^l \tag{3.31}$$

在 Fletcher 的演算法當中將疊代結果，定義成兩種型態 f-type 與 v-type ，其中 f-type 的目的爲降低目標函數 f ，而 v-type 則是降低不可行性 v 。形成 f-type 的條件爲需被過濾器所接受且眞實的目標函數減少量 Δf^k ，與預測的目標函數減少量 Δq^k 須滿足特定的準測，特定的準測如方程式(3.32)所示，因此可知形成 f-type 的條件爲方程式(3.30)、方程式(3.31)、方程式(3.32)。若 $\Delta q^k\leq 0$ 或是該次子問題(subproblem)的結果不被過濾器所接受時，則將該次疊代的主要目的放在降低不可行性 v ，稱之爲 v-type 。 v-type 成立的條件爲，若該次子問題的結果被過濾器所接受，但 $\Delta q^k<0$ ，則樣本結果加入過濾器

內。可知只有 v-type 有機會被加入過濾器內，因此在過濾器的更新過程中，並不是任意的 $\{f^k, v^k\}$ 皆會被加入至過濾器內，而是只有 v-type。餘下的情形爲當該次結果被過濾器接受且$\Delta q^k > 0$但$\Delta f^k < \zeta \Delta q^k$的情形發生，若發出此情形則將信賴區域減半，再解二次規劃問題。此種情形即不爲 f-type 或 v-type (not type)。

$$\begin{aligned} \Delta f^k &= f(\boldsymbol{\mu}_{\mathbf{X}}^{k+1}) - f(\boldsymbol{\mu}_{\mathbf{X}}^{k}) \geq \zeta \Delta q^k \text{ , 且 } \zeta \in (0,1) \\ \Delta q^k &= -\nabla f^T \cdot \mathbf{s} - \frac{1}{2}\mathbf{s}^T \mathbf{H} \mathbf{s} > 0 \end{aligned} \tag{3.32}$$

在了解 F-filter 的預備知識後，接著介紹 F-filter 的演算流程，如圖3.7爲Fletcher 等人所提出的 F-filter 演算法的流程圖 [39]，此流程圖更仔細的說明圖3.1中 F-filter 演算法的流程。首先是該次疊代過程所產生的二次規劃子問題，而解此子問題利用數學軟體 Matlab 中的 quadprog.m 做爲解二次規劃的工具(QP-Solver) [40,41]，在此使用的形式爲使

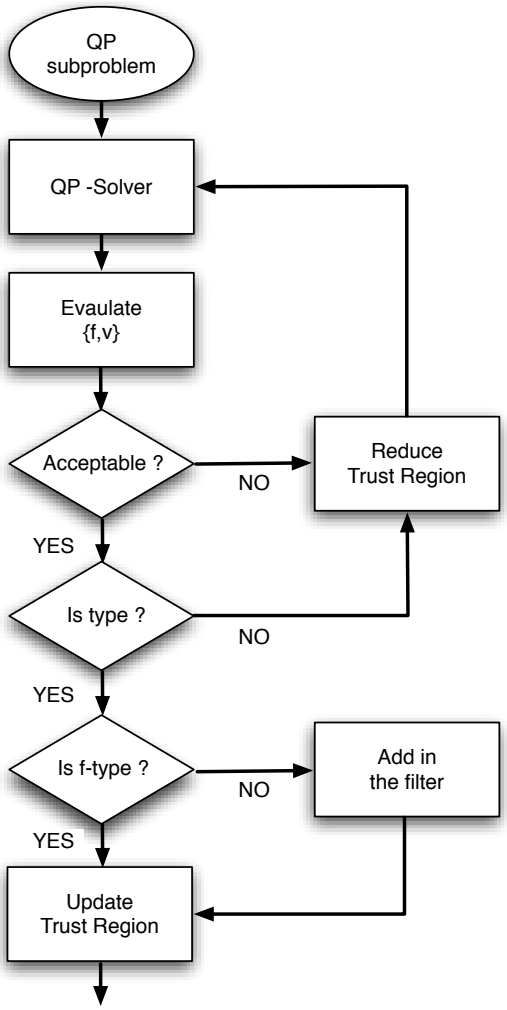

圖 3.7: F-filter 演算法流程圖

用者需提供信賴區間 (ρ)，之後得到試驗點後，計算該次疊代的結果$\{f, v\}$，判斷是否被過濾器所接受，若不接受則再解原問題但信賴區間減半，直到被過濾器所接受爲止。當被過濾器所接受後，判斷是否爲 f-type 或 v-type 其中的一種，若都不是則再解原問題且信賴區間減半，直到產生 f-type 或 v-type 爲止，此時下一次疊代的設計點也已被決定了。若是 v-type 則將該次結果加入過濾器內，最後不論是 f-type 或是 v-type 都要更新信賴區間。

Fletcher 的這個試驗的演算法還有一個需注意的是初始過濾器內的"結果" $\{f^0, v^0\}$ 的資訊該如何提供。起始的 f^0 就是此目標函數的最佳值，以全負型態而言就是最小值，此最小值是原最佳化問題中沒有拘束條件所可以找到的最小值，完全不考慮拘束條件。然而 v^0 可設定爲此演算法中最大的違反量，在實際的使用上 v^0可以訂定一個相當大的值，如 $10^4\cdots$等等 [39]。若是在考慮可靠度的最佳化問題中，則 v^0 的上限就是 1 ，意指破壞機率爲百分之百。

由於最後演算法的收斂必須爲不違反拘束條件，因此可推論每次加入過濾器的結果的趨勢會如圖3.8般的。因爲如此最終才會找到不違反拘束條件的結果。也因此可知最初始過濾器內的"結果" $\{f^0, v^0\}$ ，只要落在圖3.8的右下角，也就是每次加入過濾器的結果的趨勢的起點$(k = 0)$。這也乎應了上一段 Fletcher 給初始過濾器內的"結果" $\{f^0, v^0\}$ 制定的建議，起始的 f^0 就是儘可能的小，即使違反拘束條件也不在乎，而 v^0 可設定爲此演算法中最大的違反量，或說是 v^0 即爲找到 f^0 時的違反量。也因爲 f^0 是最小的目標函數，因此所對應的 v^0 爲最大的違反量。

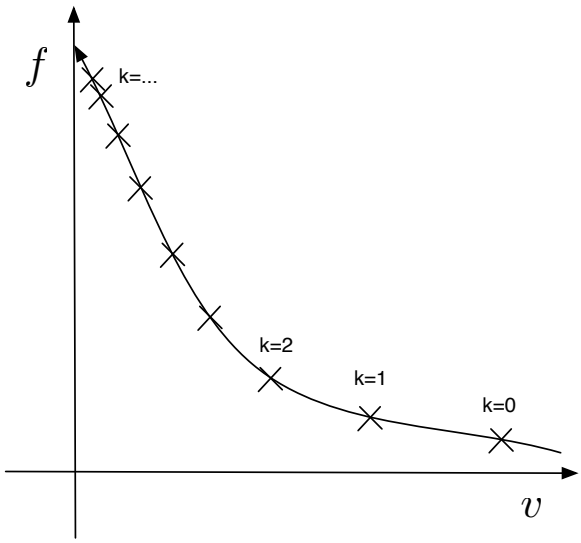

圖 3.8: F-filter 過濾器收斂趨勢圖

3.6 本章小結

本章提出了結合 AAIS 可靠度估計方法與最佳化中的 SQP 的演算流程，做爲處理 可靠度最佳化問題的方法。 AAIS 是以樣本法爲基礎的估計方法，可以計算出破壞機率與其梯度，爲了避免異常值干擾機率梯度的計算，因此增添樣本過濾器，來獲得穩健的機率梯度結果。而 SQP 則是建立在以梯度資訊的演算法，在此 SQP的最佳化演算法中以 Fletcher 等人所提出的試驗法則，來判斷該次結果是否可成爲下次疊代的起始點，試驗法則需持續進行，直到找到下次疊代的起始點爲止。爲了降低取樣的數目，會經由樣本回收機制，來回收部分或全部的樣本以供下次疊代過程使用。

第 四 章　懸臂樑工程範例

4.1　工程問題內容

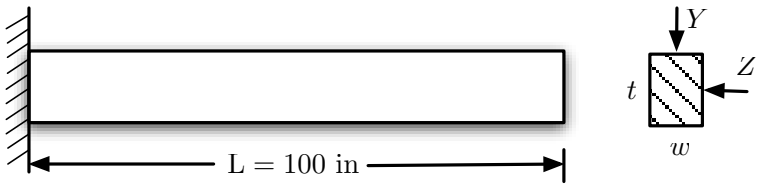

圖 4.1: 懸臂樑示意圖

考慮一根懸臂樑在樑的尖端受到垂直方向與橫向方向的力，如圖4.1所示，垂直方向負載的力大小爲一個高斯分佈，表示爲 $Y \sim \text{Normal}(1000, 100^2)$lb，而橫向方向負載的力大小也爲一個高斯分佈表示爲 $Z \sim \text{Norma}(500, 100^2)$lb。而樑的長度(L)爲 100 in。在圖4.1中 w 與 t 分別表示爲樑的寬度與厚度，單位爲 in，爲不考慮不確定因素的設計變數，然而互相的乘積爲該樑的截面積。

因爲懸臂樑在垂直與橫向遭受到彎曲力，因此懸臂樑可能破壞的情形有二:首先是懸臂樑固定端附近發生降伏，另一被定義破壞的情形爲，樑尖端的最大位移超過某一限定值 D_0 ，單位爲 in ，第一個拘束條件爲線性，而第二個拘束條件爲非線性。而目標函數視爲成本降至最低，若在懸臂樑長度與密度不變的狀況下，目標函數可訂爲將懸臂樑的截面積最小化，因此結構考慮不確定因素的最佳化問題可寫爲方程式(4.2)，其中 S_{Y} 爲一隨機變數代表降伏強度，在此以一高斯隨機變數分佈來模擬，因此寫成 $S_{\text{Y}} \sim \text{Normal}(40000, 2000^2)$psi，而 E 爲材料的楊式係數，也以一高斯隨機變數分佈來表示，因此 $E \sim \text{Normal}(2.9 \times 10^7, (1.45 \times 10^6)^2)$psi，最大可允許的破壞機率表示爲 $P_{\text{f},j}^{\text{allow}}$， $j = 1, 2$ 爲拘束條件的數目。

$$
\begin{aligned}
\min_{w,t} \quad & f(w,t) = wt \\
\text{s.t.} \quad & \Pr\left[g_1 = (\frac{600}{wt^2})Y + \frac{600}{w^2 t} Z - S_Y > 0 \right] \le P_{\text{f},1}^{\text{allow}} \\
& \Pr\left[g_2 = \frac{4L^3}{Ewt} \sqrt{\left(\frac{Y}{t^2}\right)^2 + \left(\frac{Z}{w^2}\right)^2} - D_0 > 0 \right] \le P_{\text{f},2}^{\text{allow}} \\
& 0 \le w \quad t \le 5
\end{aligned}
\tag{4.1}
$$

4.2 問題一

若問題起始的寬度與厚度爲 2.5 in 與 3 in ，最大位移量 D_0 給定爲 2.2535 in，最大可允許的破壞機率皆爲 0.00135。 再說明完工程範例的背景與內容後，接著介紹演算法內所需給定的參數。分爲演算法內容表4.1與試驗準則表4.2兩部分來描述。演算法內容表表4.1中的 Δ 選擇爲 4 ，是配合此問題的最大可允許的破壞機率，而試驗準則表4.2中的 β、γ、ζ 則是參考 Fletcher 等人的文章 [39]。

表 4.1: 問題一中的演算法參數表

No. of Samples	$\boldsymbol{\Delta}$	t(%)	ρ^0
1000	4	1	1

表 4.2: 問題一中的 F-filter 參數表

β	γ	ζ	Initial f	Initial v
0.99	0.01	0.1	0.05	0.5

經演算法計算後的結果整理如表4.3，觀察可知本文所提出的演算法結果所得之最佳目標函數比 Zissimos 等人 [42]所提出的結果好上 0.02，而兩個演算法中的拘束條件也皆 active 。

表 4.3: 結果與 Zissimos 等人之比較

Method	A Single Loop(†)	SQP with AAIS
Optimum	[2.4484,3.8884]	[2.5440, 3.7356]
f^*	9.5202	9.5034
$\star[P_1, P_2](\%)$	[0.135, 0.135]	[0.135, 0.135]

† :As reported by Zissimos et al. [42]

$\star$: validated with 10^6 MCS runs

4.3 問題二

另一情形爲當初始值寬度與厚度分別改爲 2 in 與 4 in ，而懸臂樑尖端最大位移 D_0 改爲2.5in ，而問題中所最大允許的破壞機率分別改爲 0.621%($P_{\mathrm{f},1}^{\mathrm{allow}}$) 與 0.023%($P_{\mathrm{f},2}^{\mathrm{allow}}$)，其餘條件皆與原問題相同。而使用本文所提及的演算法過程所需的參數如表4.1與4.2。

而最終經演算法計算的結果如表4.4，其中 FORM 與 SORM 的比較數據也出自於 [43]，可發現 FORM、SORM 可靠度較不準確的特性在本問題顯現出來，FORM 與 SORM 皆違反了第二個拘束條件，也就是非線性的拘束條件，然而因爲 SORM 是使用二階泰勒展開式來逼近原問題因此計算的結果比 FORM 來的更好 ，也就是破壞機率的估計更接近於要求的 0.023%。而在此問題中，本文所提出的演算法與 Rahman 等人 [43]，所獲得的結果十分的相近，目標函數與之相比，更少了 0.02，且兩個拘束條件皆 active。但本文中所使用的演算法在疊代次數較高。

表 4.4: 結果與 Rahman 等人之比較

Method	FORM	SORM	Univariate †	SQP with AAIS
No. of iteration	4	5	6	13
Optimum	[2.4530,3.7550]	[2.4580, 3.7476]	[2.4683,3.7326]	[2.4450,3.7676]
f^*	9.2109	9.2117	9.2132	9.2112
$\star[P_1, P_2](\%)$	[0.621, 0.0275]	[0.620,0.0254]	[0.618,0.0210]	[0.620,0.0233]

† :As reported by Rahman et al. [43]

$\star$: validated with 10^6 MCS runs

4.4 小結

在此次的範例中可發現 AAIS 估計可靠度的精準性，在上述的兩個範例可明顯的說明此項特色。此次的工程問題爲滿足 AAIS 的假設，即皆可順利找到控制變數，且因爲已知問題中的數學形式，並對控制變數找其反函數，所以並不討論函數計算次數，因爲本文中的函數計算次數變成是對其反函數所計並無討論的意義。而在此的範例中只使用 1000 組樣本是因爲

AAIS 是一個不需使用太多樣本的估計方法，此 1000 組是與蒙地卡羅法一般使用的十萬或是百萬組相比，顯得少了許多。但是使用 AAIS 取樣法估計破壞機率與其梯度時，並無法事先得知某問題在特定的樣本數下有如何的變異表現，因此無法在使用 AAIS 取樣法之前就決定取多少組的樣本會滿足使用者的需求，此需求指的是使用者要求的估計破壞機率的變異量，這是使用樣本法的估計策略都會遭遇到的問題，這也是本文所遭遇的一個問題。

第 五 章　針對工程模擬函數的探討

在工程上時常使用一些有限元素軟體，來輔助工程師分析問題，常見的軟體如:ansys、fluent...等等。若想針對此些軟體所建立的模型做可靠度試驗，因其函數形態未知或說是解析形態(analytic form)不存在，在此簡稱爲黑盒子函數(black-box function)，因爲無法得知黑盒子函數的數學形態，無法使用 AAIS 取樣法來做可靠度估計，因爲無法得知是否有控制變數的存在及其反函數(h_j)的形態 。此外還有雖知其解析形態，但不滿足 AAIS 取樣法之假設的問題，此章會針對以上此些問題來做一些應對與討論

5.1　工程模擬問題

本小節想處理的問題爲當函數形式不明，也就是所謂的黑盒子問題，但此些問題隱含著符合條件的控制變數。在此希望可以提出一處理流程來解決此類，函數形態未知但有符合條件的控制變數的問題。

一般而言，含有滿足假設的控制變數的函數，其數學模型可改寫成方程式(5.1)。可以發現其特色爲控制變數的次方數爲一次，也就是線性的性質。接下來的小節將會提出一連串推導與一處理流程來處理黑盒子函數且其含有滿足假設的控制變數的問題。

$$g_j(\mathbf{d},\mathbf{u}) = p_j(\mathbf{d},\bar{\mathbf{u}}) + k_j(\mathbf{d},\bar{\mathbf{u}})U_1 \tag{5.1}$$

5.1.1　方法推導

在此就通式方程式(5.1)做推導，利用 AAIS 取樣法的概念，可知當 $k_j(\mathbf{d},\bar{\mathbf{u}}) > 0$，其破壞機率及其梯度可表示爲方程式(5.2)，同理，當 $k_j(\mathbf{d},\bar{\mathbf{u}}) < 0$，破壞機率及梯度表示爲方程式(5.3)，因此藉由方程式(5.2)與方程式(5.3)，可知只要決定 $p_j(\mathbf{d},\bar{\mathbf{u}})$ 與 $k_j(\mathbf{d},\bar{\mathbf{u}})$ 就可計算一組樣本的破壞機率及其梯度，其中 $[\mathbf{d},\bar{\mathbf{u}}_\mathbf{r}]$ 表示爲第 r組 樣本。

$$\hat{P}_{\mathrm{f},j}(\mathbf{d}) \approx \frac{1}{N}\sum_{r=1}^{N}\Phi\left(\frac{p_j(\mathbf{d},\bar{\mathbf{u}}_\mathbf{r})}{k_j(\mathbf{d},\bar{\mathbf{u}}_\mathbf{r})}\right)$$
$$\nabla_\mathbf{d}\hat{P}_{\mathrm{f},j}(\mathbf{d}) \approx \frac{1}{N}\sum_{r=1}^{N}\phi(\frac{p_j(\mathbf{d},\bar{\mathbf{u}}_\mathbf{r})}{k_j(\mathbf{d},\bar{\mathbf{u}}_\mathbf{r})})\nabla_\mathbf{d}h_j \tag{5.2}$$

$$\hat{P}_{\mathrm{f},j}(\mathbf{d}) \approx \frac{1}{N}\sum_{r=1}^{N}\Phi\left(-\frac{p_j(\mathbf{d},\bar{\mathbf{u}}_\mathbf{r})}{k_j(\mathbf{d},\bar{\mathbf{u}}_\mathbf{r})}\right)$$
$$\nabla_\mathbf{d}\hat{P}_{\mathrm{f},j}(\mathbf{d}) \approx \frac{1}{N}\sum_{r=1}^{N}\phi(-\frac{p_j(\mathbf{d},\bar{\mathbf{u}}_\mathbf{r})}{k_j(\mathbf{d},\bar{\mathbf{u}}_\mathbf{r})})\nabla_\mathbf{d}h_j \tag{5.3}$$

因此由方程式(5.2)、方程式(5.3)得知，若要計算破壞機率及其梯度需求得 $p_j(\mathbf{d},\bar{\mathbf{u}}_\mathbf{r})$ 及 $k_j(\mathbf{d},\bar{\mathbf{u}}_\mathbf{r})$。因此以下將提出一套方法流程，來判斷未知函數內是否有滿足假設的控制變數，若有滿足假設的控制變數存在，則再計算 $p_j(\mathbf{d},\bar{\mathbf{u}}_\mathbf{r})$ 與 $k_j(\mathbf{d},\bar{\mathbf{u}}_\mathbf{r})$ 並估計破壞機率及其梯度。

5.1.2 方法流程

- 確認是否有滿足的控制變數
  - 指定某一隨機變數爲控制變數，對其作測試並判斷是否爲滿足假設的控制變數，其隨機變數指定的順序依變異量的大小，由大至小的順序
  - 選定三組樣本代入黑盒子函數中，樣本可表示爲 $\mathbf{o}_i = [\mathbf{d},\bar{\mathbf{u}},u_{1_i}]$，$i = 1...3$， 因此三組樣本內除了控計變數外其皆不變。 若拘束條件爲有滿足假設的控制變數時，可得方程組(5.4)

$$z_1 = g_j(\mathbf{o}_1) = p_j(\mathbf{d},\bar{\mathbf{u}}) + k_j(\mathbf{d},\bar{\mathbf{u}})u_{1,1}$$
$$z_2 = g_j(\mathbf{o}_2) = p_j(\mathbf{d},\bar{\mathbf{u}}) + k_j(\mathbf{d},\bar{\mathbf{u}})u_{1,2} \tag{5.4}$$
$$z_3 = g_j(\mathbf{o}_3) = p_j(\mathbf{d},\bar{\mathbf{u}}) + k_j(\mathbf{d},\bar{\mathbf{u}})u_{1,3}$$

  - 判斷控制變數是否存在

拘束條件爲有滿足假設的控制變數時，方程組(5.4)整理可得方程式(5.5)，因此若要測試是否滿足控制變數存在的假設時可利用方程式(5.5)，做爲驗證，即若經設

計的樣本 $\mathbf{o}_i$ 可滿足方程式(5.5)的關係，則認爲含有滿足假設的控制變數。在本文中使用三組樣本來確認，爲了確保控制變數確實存在也可多取幾組樣本。但以數值的角度，有三組就已足夠了。而在指定控制變數的順序上，從變異量較大的開始，如第2.2.7小節中所提及的。因爲變異的減少量，就是將控制變數的變異去除，可參考文獻 [5]，因此只要驗証出有符合假設的控制變數即可，不需再對其餘隨機變數在測試，即可直接進行下一步驟。

$$k_j(\mathbf{d}, \bar{\mathbf{u}}) = \frac{z_2 - z_1}{u_{1,2} - u_{1,1}} = \frac{z_3 - z_1}{u_{1,3} - u_{1,1}} \tag{5.5}$$

- 計算 $k_j(\mathbf{d}, \bar{\mathbf{u}}_\mathbf{r})$ 與 $p_j(\mathbf{d}, \bar{\mathbf{u}}_\mathbf{r})$

用方程式(5.5)可利用 $k_j(\mathbf{d}, \bar{\mathbf{u}}_\mathbf{r})$ 的量值，斷定是否有滿足的控制變數，因此方程式(5.5)也可用來計算出 $k_j(\mathbf{d}, \bar{\mathbf{u}}_\mathbf{r})$ 的量值。但由方程式(5.5)可知，需兩個特殊選擇的樣本才可決定一個 $k_j(\mathbf{d}, \bar{\mathbf{u}}_\mathbf{r})$ 的輸出，而輸入分別是$[\mathbf{d}, \bar{\mathbf{u}}, u_{1,m}]$與$[\mathbf{d}, \bar{\mathbf{u}}, u_{1,n}]$ ，$m \neq n$。若 $k_j(\mathbf{d}, \bar{\mathbf{u}}_\mathbf{r})$ 爲已知後，則 $p_j(\mathbf{d}, \bar{\mathbf{u}}_\mathbf{r})$ 也爲已知，利用方程式(5.6)，可得 $p_j(\mathbf{d}, \bar{\mathbf{u}}_\mathbf{r})$，其中的 $u_{1,r}$ 與 $z_{1,r}$ 可將先前用來計算的 $k_j(\mathbf{d}, \bar{\mathbf{u}}_\mathbf{r})$ 所取的樣本資訊再次使用以減少計算次數。

$$p_j(\mathbf{d}, \bar{\mathbf{u}}_\mathbf{r}) = z_r - k_j(\mathbf{d}, \bar{\mathbf{u}}_\mathbf{r}) u_{1,r} \tag{5.6}$$

- 獲得破壞機率與其梯度資訊

判定 $k_j(\mathbf{d}, \bar{\mathbf{u}}_\mathbf{r})$ 的正負號後，決定使用方程式(5.2)或方程式(5.3)來估計破壞機率與其梯度。

以上提供了處理工程模擬問題的方法，先計算 $k_j(\mathbf{d}, \bar{\mathbf{u}})$ 再計算 $p_j(\mathbf{d}, \bar{\mathbf{u}}_\mathbf{r})$，最後計算破壞機率與其梯度。若整體爲一個耗時的模擬時，也可用反應曲面來輔助估計 $k_j(\mathbf{d}, \bar{\mathbf{u}})$ ， 若反應曲面擬合的結果準精的話，則此法的估計結果仍如同 AAIS 取樣法有著取樣少且計算精準的特色。不論工程模擬耗時與否，估計 $k_j(\mathbf{d}, \bar{\mathbf{u}})$ 皆需兩組樣本，才能決定一組 $k_j(\mathbf{d}, \bar{\mathbf{u}})$ 的資料，但以 AAIS 取樣法只需少量樣本的特色，雖增加一倍的計算量，還是有被高度使用的價值。

5.2 加入放寬參數

爲了處理不滿足 AAIS 取樣法之假設的問題，不論是函數形式未知或是沒有滿足假設的控制變數存在，在此試著將原函數或原模擬 $g_j(\mathbf{d},\mathbf{u})$ 加入一個額外的隨機變數 U_d， U_d 可被稱爲放寬參數(slack parameter)，也就是說 U_d 爲一個參數的隨機變數。若依第三章的假設令 U_d 爲一個標準的高斯隨機變數， U_d 爲一參數的隨機變數(平均值爲 0 與標準差爲 1)。因此可將原拘束條件轉變爲方程式(5.7)，$g'_j(\mathbf{d},\mathbf{u})$ 爲加上額外變數後的函數，與一調整參數 ϵ ，只要當 $\epsilon \to 0$ 時， $g'_j(\mathbf{d},\mathbf{u})$ 就可趨近於原始問題 $g_j(\mathbf{d},\mathbf{u})$ 。

$$g'_j(\mathbf{d},\mathbf{u}) = g_j(\mathbf{d},\mathbf{u}) - \epsilon U_d \leq 0. \tag{5.7}$$

因爲將原問題轉化爲方程式(5.7)後，若 $\epsilon > 0$ 時可利用與方程式(3.10)與方程式(3.11)相同的概念來計算破壞機率與其梯度，如方程式(5.8)與方程式(5.9)。其中 $\nabla_{\mathbf{d}} g_j(\mathbf{d},\mathbf{u})$ 可利用有限差分法(finite difference)來計算，因此可表示成方程式(5.10)。同理若 $\epsilon < 0$ ，可利用與方程式(3.15)與方程式(3.16)相同的概念來計算破壞機率與其梯度，如方程式(5.11)與方程式(5.12)。由此可知，加入放寬參數的概念就是加入一個滿足 AAIS 假設的控制變數，來計算破壞機率與其梯度，下一小節則會對加入的參數做一討論。

$$\hat{P}_{j,N}(\mathbf{d}) \approx \frac{1}{N}\sum_{r=1}^{N} \Phi\left(\frac{g_j(\mathbf{d},\mathbf{u_r})}{\epsilon}\right) \tag{5.8}$$

$$\nabla_{\mathbf{d}} \hat{P}_{j,N}(\mathbf{d}) \approx \frac{1}{N}\sum_{r=1}^{N} \phi(\frac{g_j(\mathbf{d},\mathbf{u_r})}{\epsilon})\frac{\nabla_{\mathbf{d}} g_j(\mathbf{d},\mathbf{u_r})}{\epsilon} \tag{5.9}$$

$$\nabla_{\mathbf{d}} g_j \approx \lim_{\Delta\mathbf{d}\to 0} \frac{g_{j,N}(\mathbf{d}+\Delta\mathbf{d}) - g_{j,N}(\mathbf{d})}{\Delta\mathbf{d}} \tag{5.10}$$

$$\hat{P}_{j,N}(\mathbf{d}) \approx \frac{1}{N}\sum_{r=1}^{N} \Phi\left(\frac{-g_j(\mathbf{d},\mathbf{u_r})}{\epsilon}\right) \tag{5.11}$$

$$\nabla_{\mathbf{d}} \hat{P}_{j,N}(\mathbf{d}) \approx \frac{1}{N}\sum_{r=1}^{N} \phi(-\frac{g_j(\mathbf{d},\mathbf{u_r})}{\epsilon})\frac{\nabla_{\mathbf{d}} g_j(\mathbf{d},\mathbf{u_r})}{\epsilon} \tag{5.12}$$

5.3 加入放寬參數的討論

先針對 ϵ 的影響，在此分成機率與機率梯度兩部分來討論，首先是機率的部分，很理所當然的認爲 ϵ 應愈小愈好，因爲 ϵ 愈小代表原函數 $g(\mathbf{d},\mathbf{u})$ 與加上額外隨機變數的函數 $g'(\mathbf{d},\mathbf{u})$ 愈近似，那兩者計算的差距也就愈小。再觀察方程式(5.8)可知計算機率的方程式是將 N 組的樣本資料做加總並平均，而其中的 N 組資料是計算其標準高斯下的累積分佈函數的值。而在累積分佈函數內的項次 $\frac{g_j(\mathbf{d},\mathbf{u})}{\epsilon}$ 可視爲一個標準化的動作，平均值視爲 0 ，標準差爲 ϵ。平均值爲 0 是因爲額外給定的放寬參數 U_d 的平均爲 0 ，而 ϵ 則是改變 U_d 的標準差，將原來 U_d 的標準差 1 乘上 ϵ 倍，或該說 U_d 是平均值 0 標準差爲 ϵ 的高斯分佈。再次回顧方程式(5.8)，當 ϵ 越小時，會發現標準高斯下的累積分佈函數值會愈是 0 或是 1 ，因爲 $\frac{g_j(\mathbf{d},\mathbf{u})}{\epsilon}$ 的值不是極大就是極小，此結果如同蒙地卡羅法一般。反之，若 ϵ 越大時，因爲 $\frac{g_j(\mathbf{d},\mathbf{u})}{\epsilon}$ 的值趨近於 0 ，雖然可以降低變異，但也因此標準高斯下的累積分佈函數值會很接近 $\frac{1}{2}$。因此在 ϵ 在破壞機率上的影響，可說是當 ϵ 越小，越接近蒙地卡羅法，效率越差，若ϵ 越大則造成估計上的不準。

在機率梯度方面，由方程式(5.9)、方程式(5.12)可知 $\nabla_{\mathbf{d}} g_j(\mathbf{d},\mathbf{u})$ 爲函數的梯度項次，而 $\phi(\frac{g_j(\mathbf{d},\mathbf{u})}{\epsilon})/\epsilon$ 會是一個調整比例的項次，因而 ϵ 的選擇會影響梯度計算的結果。圖5.1是比例的項次的等高線圖，橫軸爲比例的項次中的 $g_j(\mathbf{d},\mathbf{u})$，而縱軸爲 ϵ，可以發現圖5.1以縱軸呈現對稱的情形，因爲是兩個偶函數的四則運算。因此也就是說，不論 $g_j(\mathbf{d},\mathbf{u})$ 的正負或說是不論拘束條件的破壞與安全。當縱軸 ϵ 增加時，比例項次也隨著增加，而橫軸 $g_j(\mathbf{d},\mathbf{u})$ 增加時，比例項次卻隨著減少。因此若 ϵ 極小或 $g_j(\mathbf{d},\mathbf{u})$ 極大時，此比例項次很有可能是極爲接近 0 的數值。可知 ϵ 與 $g_j(\mathbf{d},\mathbf{u})$之間有個比例(scale)的關係，又 $g_j(\mathbf{d},\mathbf{u})$ 的數值大小是與原問題有關的，因此不改變 $g_j(\mathbf{d},\mathbf{u})$ 的比例，只考慮如何制定 ϵ 的大小。由以上所述可知 ϵ 的結論，若選擇極小的 ϵ 會造成有準確的可靠度，但造成效率的低落且可能會造成機率梯度的結果成爲零向量。因此 ϵ 的選擇是個困難的決定。最後以二個數學的範例來說明當問題無法滿足第3.2.1小節中的假設時，以第5.2小節中的方法來處理該問題，討論所解出的結果。

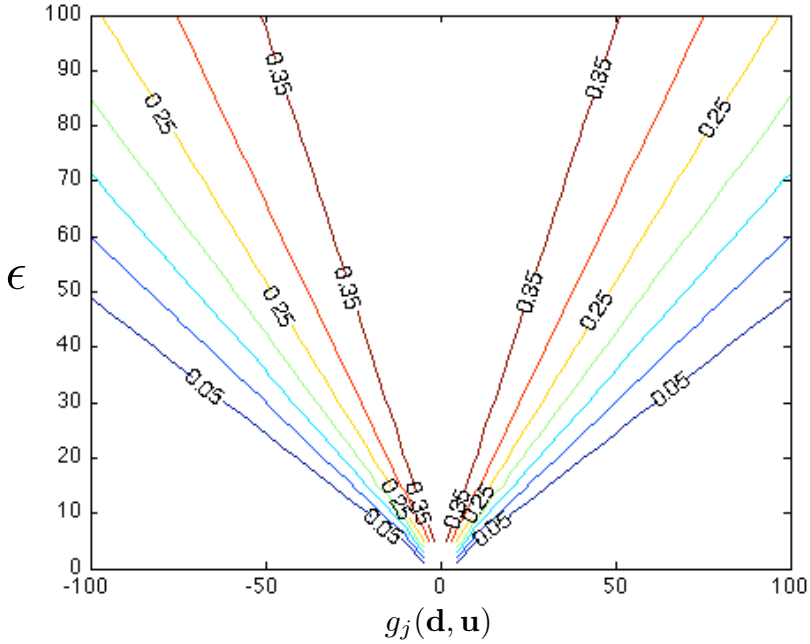

圖 5.1: 比例項次的等高線圖

本文當中放寬參數使用標準的高斯分佈，除了，爲與先前第三章敘述一致的原因之外，還有另一理由爲希望借助高斯分佈左右對稱的特色，也就是藉由平均值 0 來區分破壞和安全兩種情形，並對等處理破壞與安全的情形。這也是爲甚麼使用平均值爲 0 的標準高斯分佈的原因，但函數的遞增遞減情形不一定會相同，因此使用高斯分佈也還具有討論的空間。因此在放寬參數使用的討論上，可就兩方面來做探討，放寬參數的隨機分佈選擇，以及隨機分佈參數的制定。換句話說隨機分佈不一定爲高斯分佈，高斯隨機分佈中的平均值也不一定爲 0，因此放寬參數的使用還有許多值得探討的空間。

使用放寬參數可以處理不滿足 AAIS 取樣法假設的問題，但卻無法提升取樣的效率，因爲變異降低的原因，來自於少了控制變數的變異影響所造成的。因此若是以放寬參數爲控制變數時，則變異量並無明顯減少的跡象，因爲減少的是放寬參數的變異量，而原問題中並無放寬參數的出現，所以變異量並沒有降低。若有些微的降低，則是因爲 AAIS 取樣法使用累積密度函數與機率密度函數來估計破壞機率與其梯度，所得到的資訊不爲 0 或 1 ，而是 0 至 1 之間的資訊 ，所以變異量可能會比蒙地卡羅法稍稍低些。

5.4 敏感度分析

在介紹了典型的 AAIS 取樣方法與加入放寬參數的 AAIS 取樣法後，針對上述的方法與蒙地卡羅來做一個敏感度分析的比較。以某一數學形式已知的拘束條件爲例，如方程式(5.13)所示。其破壞機率計算的結果如圖5.2所示，橫軸爲取樣的樣本數，但需注意的是，其，是以十爲底的對數爲單位，而縱軸則爲破壞機率，而此敏感度分析方法是以不同的樣本數分別做破壞機率的估計，並重複 5000 次取其平均值，爲使用該樣本數所計算破壞機率的代表。其中實線爲蒙地卡羅的結果，虛點線則爲 AAIS 取樣法的結果，虛線爲 AAIS 取樣法並融入放寬變數，也就是第5.2小節所提及的內容。

$$g(\mathbf{d}, \mathbf{U}) = (d_2 + U_2)^2(d_1 + U_1) - 20 > 0 \tag{5.13}$$

若以蒙地卡羅百萬點的計算結果作爲眞實的破壞機率的依據，可以發現典型 AAIS 取樣法的計算結果相當接近眞實的破壞機率，並且在樣本數不多的情形之下依然有這種表現，約在 900 組樣本時就可達到蒙地卡羅百萬組樣本的結果表現，符合初始使用 AAIS 取樣法的初衷。然而在融入放寬變數的 AAIS 取樣法在表現上，與蒙地卡羅的整體表現相當的接近，起初的起伏較大，樣本數過萬了之後才有較穩健的表現，也呼應了第2.2.7小節中所提及的結果會與蒙地卡羅法相似。

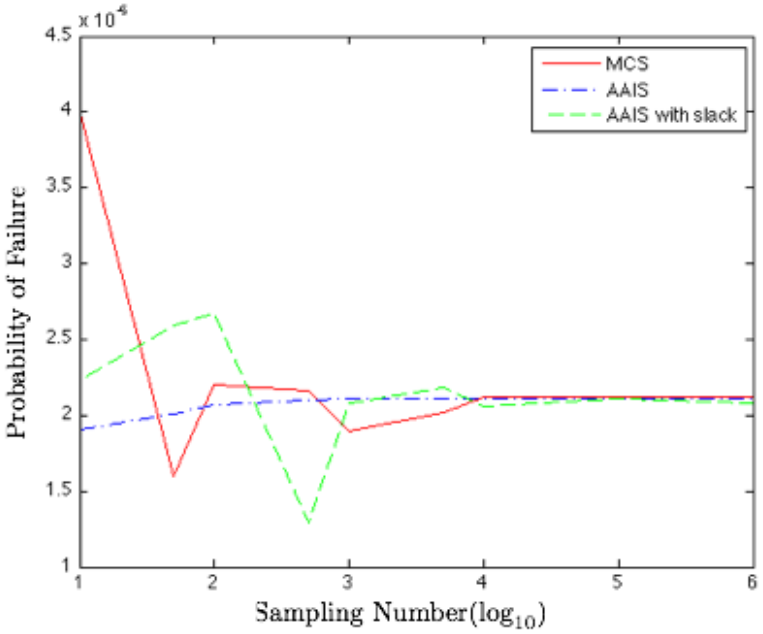

圖 5.2: 破壞機率

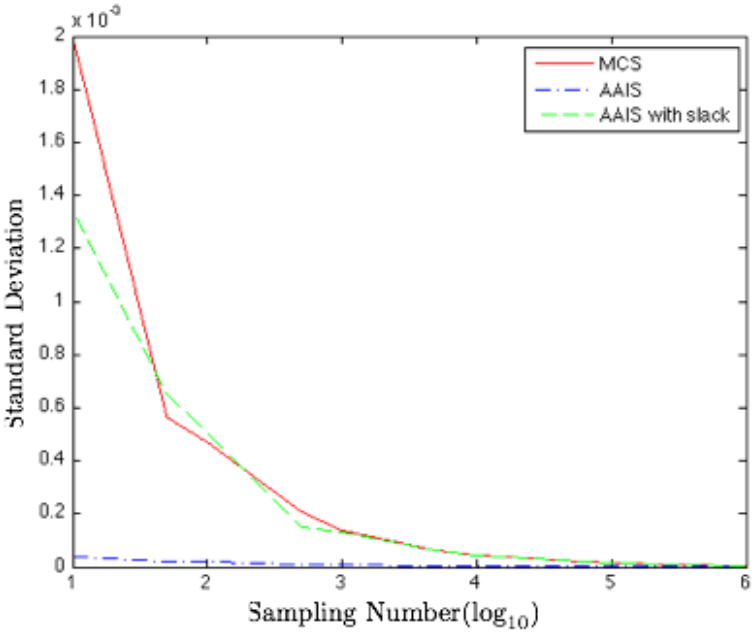

圖 5.3: 破壞機率

而在 5000 次計算破壞機率的變異結果，如圖5.3所示，橫軸依然爲取樣的樣本數，並以十爲底的對數爲單位，而縱軸則爲標準差。由圖5.3可以發現，使用典型 AAIS 取樣法的標準差比起使用蒙地卡羅法的標準差低了許多，蒙地卡羅法使用上萬組的變異與典型 AAIS 取樣法使用 100 組樣本的變異相接近，也就是說典型的 AAIS 取樣法可以用較少的樣本就有蒙地卡羅大量樣本的精確性。然而在融入放寬變數的 AAIS 取樣法的表現，除了初期的標準差比蒙地卡羅法較佳之外，其餘部分的表現與蒙地卡羅法相當。這又呼應的在第2.2.7小節中所提到的，降低變異的來源，來自於少了控制變數的變異影響所造成的，因此若是以放寬參數爲控制變數時，則變異量並沒有減少，因爲減少的是放寬參數的變異量，而原問題中並無放寬參數的出現，所以變異量並沒有降低，也就是與蒙地卡羅的結果近似，此現象也反應在計算破壞機率的圖5.2中。

由此可知 AAIS 取樣法是一種使用少量的樣本並且具有高準確性的一種估計方法，而加入放寬參數的 AAIS 取樣法則是一種類似蒙地卡羅的估計方法，雖然其效率不高，但其優點爲可以有效的計算破壞機率梯度，只要在 ϵ 制定適當的前提之下。

5.5 多個最大可能破壞點的問題

考慮處理如方程式(5.14)中的問題，問題中包含了兩個高斯隨機變數，初始的設計點爲 $[\mu_{X_1}, \mu_{X_2}] = [-2, 0]$，所對應的標準差皆爲 1。而所能接受的最大破壞機率爲 6%。

$$
\begin{aligned}
&\min_{\boldsymbol{\mu}_{\mathbf{X}}} f(\boldsymbol{\mu}_{\mathbf{X}}) = -\mu_{X_2} \\
&\text{s.t.} \quad \Pr\left[g = X_1^2 + X_2^2 - \left(3 + \cos\left(3\arctan\left(\frac{X_2}{X_1}\right)\right)^2\right) > 0\right] \leq P_{\mathrm{f}}^{\mathrm{allow}} \\
&X_i \sim N(\mu_{X_i}, \sigma_{X_i}), \ i = 1, 2
\end{aligned}
\tag{5.14}
$$

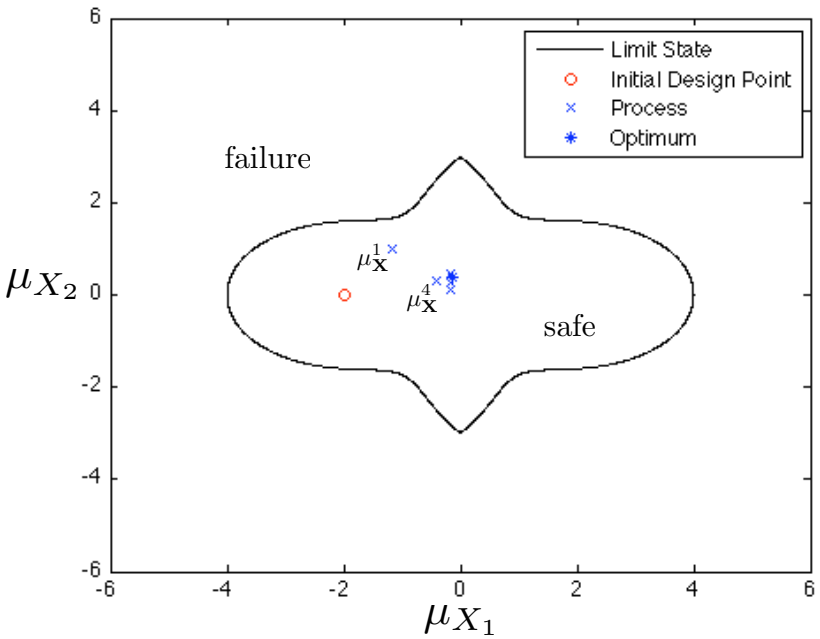

圖 5.4: 具有多個最大可能破壞點問題的表示圖

參數訂定

再說明完範例的背景與內容後，接著介紹演算法內所需給定的參數。分爲演算法內容表5.1與試驗準則表5.2兩部分來描述。演算法內容表表5.1中的 Δ 選擇爲 4 ，是配合此問題的最大可允許的破壞機率，而試驗準則表5.2中的 β、γ、ζ 則是參考 Fletcher 等人的文章 [39]。

表 5.1: 多個最大可能破壞點問題中的演算法參數表

No. of Samples	ϵ	$\boldsymbol{\Delta}$	t(%)	ρ^0
500	0.7	4	1	1

表 5.2: 多個最大可能破壞點問題中的 F-filter 參數表

α	γ	ζ	Initial f	Initial v
0.99	0.01	0.1	0.05	0.5

結果與分析

此問題很明顯的無法直接用 AAIS 方法來求得破壞機率與其梯度，因此以第5.2小節的加入放寬參數方法，來求解破壞機率與梯度。最後經由演算法獲得的最佳值爲 -0.376，而最佳解 $[-0.135, 0.376]$，而共使用了 3451 個樣本， 將最佳解以蒙地卡羅法取十萬點來確認該設計點的破壞機率爲 6.2%。由圖5.4可知其演算疊代過程，此外這數學問題還有一特色有多個最大破壞點(MPP)，無法直接用進階一階二次可靠度方法來求解，用蒙地卡羅法配合 SQP (Matlab 中的fmincon.m，見文獻 [44])會發生提供錯誤梯度資訊，造成演算法無法收斂。而在此提出的演算法可順利的在十個疊代過程找到最佳解。

在此觀察表5.3中的最後三個疊代結果，可以發現最後二次疊代因設計變數相當靠近，因此回收了大量的樣本使得，最後一次疊代僅取 55 組樣本即可。而再觀察倒數第三次與倒數第二次，卻發現設計變數雖靠近，但回收的樣本卻少的多。這是因爲在第三次與倒數第二次的疊代過程中，多次的出入 F-filter 演算法內的過濾器，多次的將信賴區間減半，直到計算出下回疊代的起始值。因爲每次出入過濾器需再多取樣本，而多次出入過濾器則造成樣本數增加。因此在第三次與倒數第二次的疊代過程中，不該說回收樣本數減少，該說是 F-filter

表 5.3: 多個最大可能破壞點問題的疊代過程

Iteration	Design Variables		No. of new samples
1	-2.000	0.000	500
2	-1.159	1.000	127
3	-0.159	0.460	412
4	-0.409	0.314	378
5	-0.159	0.127	437
6	-0.161	0.252	155
7	-0.147	0.377	445
8	-0.146	0.376	497
9	-0.134	0.376	445
10	-0.134	0.376	55

演算法內的過濾器造成取樣數額外的增加。在最極端的情形中會有某次疊代過程使用的樣本數，比每次疊代使用的樣本數還高，以此範例來說就是設定的 500 次。但也有可能因步伐很小而不用取任何一組樣本。

5.6 可靠度最佳化數值問題

第二個例子是一個可靠度最佳化的測試問題，許多論文也有對其作測試 [45–47]。在此問題中 X_1 與 X_2 爲高斯分佈的設計變數，初始值爲 $\mathbf{X} = [5, 5]$，而標準差皆爲 0.3 ，而最大允許的破壞機率皆爲 0.0013。

$$
\begin{aligned}
\min_{\boldsymbol{\mu}_{\mathbf{X}}} \quad & f(\boldsymbol{\mu}_{\mathbf{X}}) = \mu_{X_1} + \mu_{X_2} \\
\text{s.t.} \quad & \Pr\left[g_1 = X_1^2 X_2 - 20 > 0\right] \le P_{\text{f},1}^{\text{allow}} \\
& \Pr\left[g_2 = 1 - \frac{(X_1 + X_2 - 5)^2}{30} - \frac{(X_1 - X_2 - 12)^2}{120} > 0\right] \le P_{\text{f},2}^{\text{allow}} \\
& \Pr\left[g_3 = X_1^2 + 8X_2 - 75 > 0\right] \le P_{\text{f},3}^{\text{allow}} \\
& X_i \sim N(\mu_{X_i}, 0.3^2),\ i = 1, 2
\end{aligned}
\tag{5.15}
$$

參數訂定

再說明完範例的內容後，接著介紹演算法內容與試驗準則所需給定的參數。
演算法內容的參數見表5.4，而試驗準則的參數則見5.5。演算法內容表表5.4中的 Δ 選擇爲 4 ，是配合此問題的最大可允許的破壞機率，而試驗準則表5.5中的 β、γ、ζ 則是參考 Fletcher 等人的文章 [39]。

表 5.4: 可靠度最佳化問題中的演算法參數表

No. of Samples	ϵ	$\boldsymbol{\Delta}$	t(%)	ρ^0
100	0.001	4	1	1

表 5.5: 可靠度最佳化問題中的 F-filter 參數表

α	γ	ζ	Initial f	Initial v
0.99	0.01	0.1	0.05	0.5

結果與分析

此問題中的第二個拘束條件，不滿足 AAIS 問題的假設，找不到符合條件的控制變數，因此使用第5.2小節中的方法，加入放寬參數，來處理。在圖5.5可知此演算法的疊代過程，而表5.6可得的此問題的結果並與其它方法比較。由表中可知除了第二個拘束條件與相比較之方

表 5.6: 結果與 Du 等人及 Chan 等人之比較

Method	SORA(†)	SLP(‡)	SLP with AAIS
Optimum	[3.4409, 3.2909]	[3.4391,3.2827]	[3.4127, 3.3703]
f^*	6.7318	6.7336	6.8027
No. of iterations	16	4	11
No. of Func. Evals	520	44	1674
$\star[P_1, P_2, P_3](\%)$	[0.14, 0.11, 0]	[0.13, 0.13, 0]	[0.13, 0.03, 0]

† : as reported by Du et al. [46],‡:as reported by Chan et al. [45] $\star$: validated with 10^6 MCS runs

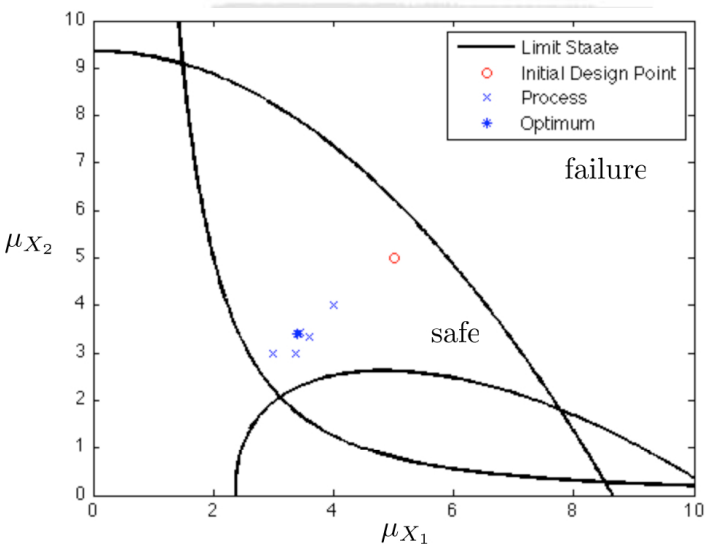

圖 5.5: 可靠度最佳化問題的疊代過程

法有較大的出入之外，其餘拘束條件皆大致相同。因第二個拘束條件使用加入放寬參數的方法，而導致估計上出了誤差。與 Chan 等人 [45]相比較時可發現第二個拘束條件應 active ，但用本文中的方法卻沒有 active，造成的原因很大的原因是 ϵ 沒有訂定妥當，造成可靠度計算上的誤差或是計算機率梯度上的誤差。

5.7 本章小結

利用第5.1小節所介紹的方法，可有效的處理函數形式不明但含有滿足假設的控制變數之問題，隨然計算量會有所增加，但以 AAIS 取樣法只需少量樣本的特色，雖增加計算量，還是有被高度使用的價值。

在本章中使用了第5.2小節中的方法來處理此章所面對的問題，在多個最大破壞點的問題當中發現用“適當”的 ϵ 眞的可有效的解決當不知問題形式或是沒有滿足假設的控制變數時的問題。在第二個問題中就發現了不”適當”的 ϵ 會提供演算法一些不正確的資訊，如太大的 ϵ 會造成計算機率值的錯誤，若太小則可能造成梯度值趨近於 0 ，因此 ϵ 的調整實爲此方法之關鍵所在。

在本文內還無一個有效的方來制法 ϵ，只能在進行演算法時，以第一次取樣的樣本爲基準，調整 ϵ 的大小，使其與第一次取樣的樣本的蒙地卡羅結果相近，取其 ϵ 的上限，作爲目前 ϵ 訂定的方法。

第 六 章　結論與建議

本研究針對考慮可靠度的最佳化問題提出了一個演算流程，包括使用 AAIS 取樣法的概念來計算破壞機率與其梯度，使用樣本過濾器確保梯度資訊不受異常值的干擾，再以 F-filter 作爲試驗準則，得到下一疊代的起始點，最後將可回收樣本回收，做爲下一次取樣的取本，因此下一次疊代只需在上一次疊代過程中還未取樣的區域進行取樣即可。

6.1　研究貢獻

茲將本研究的結論與貢獻總結如下:

1. 將 AAIS 取樣法中的控制變數延伸至帶有隨機的設計變數
   AAIS 取樣法的特色爲一種將樣本取在拘束條件邊界上，即極限狀態下的一種方法，因此可達到高精準度且較少樣本數的優勢，但也因此被某些假設所限制。

2. 提出一個穩健的方法計算破壞機率梯度
   AAIS 取樣法提出一個具備完整數學背景，並可處理機率梯度的問題。但爲了使機率梯度不受異常值所影響，而增設了樣本過濾器，以獲得更穩健的機率梯度。

3. 將 AAIS 取樣法延伸至工程模擬函數
   提出一個延伸 AAIS 取樣法的概念，加入一個放寬參數，試著處理當典型的 AAIS 取樣法所不能處理的問題。

4. 提出樣本回收的機制，使得樣本法更有效率
   利用樣本回收機制，期許減少總體使用的樣本數。

5. 提出一個以取樣法與 SQP 結合的演算法，處理可靠度最佳化的問題

6. 在演算中使用有別於懲罰函數或障礙函數的方法
   在演算法中使用 F-filter 做爲試驗準則，判斷每次疊代結果是否可被接受爲下一疊代的起始點。

7. 完成一個通用的 MATLAB 程式碼，方便後人的使用
   在最佳化中很重要的就是自動化的概念，因此撰寫一個 MATLAB 程式碼，僅需依程式要求，輸入所需的資料，即可獲得結果。

由以上的結論，可知 AAIS 取樣法可說是本文所提出的演算法的核心，因此下列給與的建議以突破 AAIS 取樣法的假設爲主軸。

6.2 未來研究方向

茲提出以下幾點建議，做後續研究之參考:

1. 在 AAIS 取樣法的假設當中，控制變數與各個隨機變數間需爲獨立，但要找到反函數存在的控制變數實已不易，因此若能打破此項假設，將再提升使用的廣度。

2. 在使用樣本法時，最常被討論的就是多少的樣本數才足以滿足設計者的需求。文獻 [48]，Shooman 在1968 年與文獻 [49]，Ayyub 在 1985年，分別有對蒙地卡羅的樣本數做一探討。因此若能對 AAIS 取樣法來做樣本數所用的探討，則更能了解 AAIS 取樣法的特性與其極限。

3. 探討 ϵ 的訂定，由第5.3小節可知當 ϵ 極小可得到類似蒙地卡羅的結果，但極有可能造成破壞機率梯度成零向量。因此 ϵ 的訂定極爲關鍵，往後或許可試著使用函數來模擬 ϵ 或許也不失爲一種方法。

4. 放寬變數分佈的選擇與其參數制定，這方面是一個很值得嘗試的方向，甚至可將放寬參數轉換至設計變數當中使其不爲一個參數形式的隨機變數。

5. 當遇到耗時的模擬問題或是高成本的試驗問題時可加入反應曲面來處理，降低成本或時間。

參 考 文 獻

[1] P. Y. Papalambros and D. F. Wilde, *Principles of Optimal Design.* Cambridge, 2000.

[2] K. N. Otto and K. L. Wood, *Product Design Techniques in Reverse Engineering and New Product Development.* New Jersey: Prentice Hall, 2001.

[3] M. Allen and K. Maute, "Reliability-based shape optimization of structures undergoing fluid-structure interaction phenomeena," *Computer Methods in Applied Mechanics and Engineering*, vol. 194, pp. 3472–3495, 2005.

[4] L. Gu, R.-J. Yang, C. Tho, M. Makowski, O. Faruque, and Y. Li, "Optimization and robustness for crashworthiness of side impact," *International Journal of Vehicle Design*, vol. 26, no. 4, pp. 348–360, 2001.

[5] A. Haldar and S. Mahadevan, *Probability, Reliability and Statistical Methods in Engineering Design.* New York: John Wiley & Sons, 2000.

[6] L. Huyse, S. Padula, R. Lewis, and W. Li, "Probabilistic approach to free-form airfoil shape optimization under uncertainty," *AIAA Journal*, vol. 40, no. 9, pp. 1764–1772, 2002.

[7] S. Yoshikazu, N. Hirotaka, and T. Tetsuzo, *Theory of Multiobjective Optimization.* Academic Press, INC, 1996.

[8] J. Royset and E. Polak, "Implementable algorithm for stochastic optimization using sample average approximations," *Journal of Optimization theory and applications*, vol. 233, no. 1, pp. 157–184, July 2004.

[9] J. Royset and E. Polak, "Reliability-based optimal design using sample average approximations," *Probabilistic Engineering Mechanics*, vol. 19, no. 4, pp. 331–343, 2004.

[10] C. Cornell, "Bounds on reliability of structural systems," *American Society of Civil Engineers Proceedings, Journal of the Structural Division*, vol. 93, no. ST1, pp. 171–200, 1967.

[11] R. Rackwitz, "Practical probability approach design," *Comite European du Beton*, no. 112, 1976.

[12] A. Hasofer and N. Lind, "Exact and invariant second-moment code format.," *Journal of the Engineering Mechanics Division*, vol. 100, no. EM1, pp. 111–121, 1974.

[13] M. Hohenbichler and R. Rackwitz, "Non-normal dependent vectors in structural safety," *ASCE Engineering Mechanics Division*, vol. 107, no. 6, pp. 1227–1238, 1981.

[14] X. Chen and N. Lind, "Fast probability integration by three-parameter normal tail aproximation.," *Structural Safety*, vol. 1, pp. 269–276, 1983.

[15] R. Rackwitz and B. Fidssler, "Structural reliability under combined random load sequences.," *Computers and Structures*, vol. 9, no. 5, pp. 484–494, 1978.

[16] B. Fiessler, H.-J. Neumann, and R. Rackwitz, "Quadratic limit states in structural reliability," *ASCE Engineering Mechanics Division*, vol. 105, no. 4, pp. 661–676, 1979.

[17] K. Breitung, "Asymptotic approximations for multinormal integrals.," *Journal of Engineering Mechanics*, vol. 110, no. 3, pp. 357–366, 1984.

[18] M. Hohenbichler and R. Rackwitz, "Fiirst-order concepts in system reliability.," *Structural Safety*, vol. 1, no. 3, pp. 177–188, 1983.

[19] O. Ditlevsen, "Generalized second moment reliability index," *Journal of Structural Mechanics*, vol. 7, no. 4, pp. 435–451, 1979.

[20] L. Tvedt, "Distribution of quadratic forms in normal space-application to structural reliability.," *Journal of Engineering Mechanics, ASCE*, no. 6, 1990.

[21] J. Halton, "Retrospective and prospective survey of the monte carlo method," *SIAM Review*, vol. 12, no. 1, pp. 1–63, 1970.

[22] J. Turner, H. Wright, and R. Hamm, "A monte carlo primer for health physicists," *Health Physics*, vol. 48, no. 6, pp. 717–733, 1985.

[23] H. Rief, E. Gelbard, R. Schaefer, and K. Smith, "Review of monte carlo techniques for analyzing reactor perturbations," *Nuclear Science and Engineering*, vol. 92, no. 2, pp. 289–297, 1986.

[24] B.-M. Ayyub and C.-Y. Chia, "Generalized conditional expectation for structural reliability assessment," *Structural Safety*, vol. 11, pp. 131–146, 1992.

[25] L.-D. Jay, *Probability and Statistics.* Thomson, 6th ed., 2004.

[26] A. Harbitz, "An efficient sampling method for probability of failure calculation," *Structural Safety*, vol. 3, no. 2, pp. 109–115, 1986.

[27] A. Karamchandani, P. Bjerager, and A. Cornell, "Adaptive importance sampling.," *Proceedings International Conference on Structural Safety and Reliability, San Francisco*, pp. 855–862, 1989.

[28] B.-M. Ayyub and A. Haldar, "Practical reliability techniques," *Journal of Structure Engineering*, vol. 110, pp. 1707–1784, 1984.

[29] G.-J. White and B.-M. Ayyub, "Reliability methods for ship structures," *ASME Naval Engineering Journal*, vol. 97, pp. 86–96, 1985.

[30] R.-E. Melchers, *Structural Reliability Analysis and Prediction.* Chichester, UK: Ellis Horwood, 1987.

[31] G. Fishman, *Monte Carlo : concepts, algorithms, and applications.* New York: Springer, 1996.

[32] E. Hofer, "Sensitivity analysis in the context of uncertainty analysis for computationally intensive models," *Computer Physics Communications*, vol. 117, no. 1-2, pp. 21–34, 1999.

[33] A. Karamchandani, P. Bjerager, and C. Cornell, "Adaptive importance sampling," in *Proceedings of the 5th International Conference on Structural Safety and Reliability*, vol. Part II, pp. 855–862, Aug. 7-11 1989.

[34] Y. Wu, "Adaptive importance sampling AIS-based system reliability sensitivity analysis method," in *Proceedings of the IUTAM Symposium, Jun 7-10 1993*, p. 550, 1993.

[35] J. Tu, K. Choi, and Y. H. Park, "Design potential method for robust system parameter design," *AIAA Journal*, vol. 121, no. 4, pp. 557–564, 2001.

[36] J. Tu and K. Choi, "A new study on reliability-based design optimization," *ASME Journal of Mechanical Design*, vol. 121, no. 4, pp. 557–564, 1999.

[37] X. Chen, T. K. Hasselman, and D. J. Neil, "Reliability based structural design optimization for practical applications," *In Proceedings of the 38th AIAA/ASME/ASCE/AHS/ASE Structures, Structural Dynamics, and Materials Conference. Part 4 (of 4)*, vol. IMM-TR-2002-12, pp. 2724–2732, April 7-10 1997.

[38] R. Yang, C. Chuang, L. Gu, and G. Li, "Numerical experiments of reliability-based optimization methods," *In proceeding of the 45th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, vol. 7, pp. 5393–5405, 2004.

[39] R. Fletcher, S. Leyffer, and P.-L. Toint, "On the global convergence of a filter-SQP algorithm," *Journal of Optimization*, vol. 13, no. 1, pp. 44–59, 2002.

[40] T. Coleman and Y. Li, "A reflective newton method for minimizing a quadratic function subject to bounds on some of the variables," *Journal on Optimization ,SIAM*, no. 4, pp. 1040–1058, 1996.

[41] P. Gill, W. Murray, and M. Wright, *Practical Optimization.* London, UK: Academic Press, INC, 1981.

[42] L. Jinghong, P. M. Zissimos, and N. Efstratios, "A single-loop approach for system reliability-based design optimization," *Journal of Mechanical Design*, no. 4, pp. 1215–1224, 2007.

[43] S. Rahman and D. Wei, "Design sensitivity and reliability-based structural optimization by univariate decomposition," *Struct Multidisc Optim*, pp. 245–261, 2008.

[44] *MATLAB optimization toolbox: User's guide.* Massachusetts: The MathWorks Inc., 2009.

[45] K.-Y. Chan, S. Skerlos, and P. Papalambros, "An adaptive sequential linear programming algorithm for optimal design problems with probabilistic constraints," *Journal of Mechanical Design*, vol. 129, no. 2, pp. 140–149, 2007.

[46] X. Du and W. Chen, "Sequential optimization and reliability assessment method for efficient probabilistic design," *Journal of Mechanical Design*, vol. 126, no. 2, pp. 225–233, March 2004.

[47] B. Youn, K. Choi, and Y. Park, "Hybrid analysis method for reliability-based design optimization," *Journal of Mechanical Design, Transactions of the ASME*, vol. 125, no. 2, pp. 221–232, 2003.

[48] M. Shooman, *Probability Reliability: An Engineering Approach.* New York: McGraw-Hill, 1968.

[49] B. Ayyub and A. Haldar, "Decision in construction operation," *Journal of the Construction Engineering and Management Division, ASCE*, no. 4, pp. 343–357, 1985.

附錄一：中英對照表

acceptable 接受

adaptive important sampling 重要調整取點法

conditional expectation 條件期望值

control variable 控制變數

design variable 設計變數

deterministic design variable 定值設計變數

dominate 支配

feasible space 可行解空間

fitting 擬合

ill-condition 病態條件

important sampling 重要取點法

limit state 極限狀態

most probable point 最大可能破壞點

negative null form 全負型態

penalty functin 懲罰函數

QP subproblem 二次規劃子問題

radius-based important sampling 半徑取點法

random variable 隨機變數

reliability-based optimum design 可靠度最佳化

reliability index 可靠度指標

slack parameter 放寬參數

trial point 試驗點

trust region 信賴區間

uncertainty 不確定因素

variance 變異

variance reduction techniques 減少變異的取點技法

自　　傳

- 姓名：許凱勛

- 籍貫：台中市

- 求學經歷：

  – 國立成功大學機械系 (2004∼2007)

  – 國立成功大學機械所 (2007∼2009)

- 著作： K.-S. Hsu and K.-Y. Chan, “A Filter-Based Sample Average SQP for Optimization Problems with Highly Nonlinear Probabilistic Constraints”, Proceedings of the ASME International Design Engineering Technical Conference, San Diego, CA, Auguest 30- September 2, 2009