

ЛЕКЦІЯ 8

Оптимізація конструкції ущільнення.

Оптимізація у програмному комплексі ANSYS

Програма ANSYS [7] має у своєму розпорядженні можливості не тільки для розрахунку конструкції, але і для знаходження її оптимального варіанта (оптимального проекту). Оптимальним є проект, який відповідає всім вимогам, які ставляться, і має мінімальні значення певних показників, таких, як вага, площа поверхні, об'єм, напруження, власні частоти і т.п.

У ANSYS доступні методи і засоби оптимізації. Необхідно відразу зазначити різницю між ними. Методи оптимізації (methods) забезпечують оптимізацію цільової функції шляхом варіювання вхідних параметрів. Засоби оптимізації (tools) забезпечують отримання декількох наборів вихідних параметрів (цільова функція, змінні стану) при зміні вхідних параметрів за заданим законом, оптимізацію цільової функції вони не виконують.

Перед описом методики оптимізації в ANSYS, необхідно визначити деякі важливі терміни.

Змінні проекту (design variables) – це параметри, які змінюються з метою знаходження оптимального проекту. Для змінних проекту вказуються обмеження – мінімальне і максимальне значення. Ці значення визначають діапазон зміни змінних проекту. Змінними проекту звичайно є геометричні параметри, такі, як довжина, товщина, діаметр або координати точок. Змінні проекту можуть набувати тільки додатних значень.

Змінні стану (state variables) – це параметри, які служать обмеженнями для проекту. Вони також називаються залежними змінними. Як правило, вони є параметрами відгуку, що є функціями змінних проекту. Змінні стану можуть бути обмежені максимальним і мінімальним значеннями або мати тільки одне з цих обмежень. Прикладами змінних проекту є напруження, власні частоти, деформації і т.п. Проте змінна стану необов'язково повинна бути обчислюваною величиною, як змінна стану може бути вказаний будь-який параметр.

Цільова функція (objective) – це залежна змінна, яку вимагається мінімізувати. Вона повинна бути функцією змінних проекту, тобто зміна значень змінних проекту повинна змінювати значення цільової функції. У оптимізаційній задачі може бути визначена тільки одна цільова функція.

Змінні проекту, змінні стану і цільова функція узагальнено називаються змінними оптимізації (optimization variables). Користувач повинен вказати, які параметри в моделі є змінними проекту, змінними стану і цільовою функцією.

Набір параметрів проекту, або проект (design set, design), – це набір значень параметрів, що являють собою яку-небудь конфігурацію моделі. Як правило, набір параметрів проекту характеризується значеннями змінних оптимізації, проте в нього включаються всі параметри моделі, у тому числі і ті, які не є змінними оптимізації.

Можливий проект (feasible design) – це проект, який задовольняє всі вказані обмеження – змінні стану і змінні проекту. Якщо хоча б одне з обмежень не виконується, проект називається неможливим (infeasible design). Найкращий проект (best design) – це той, який задовольняє всі обмеження і забезпечує мінімальне значення цільової функції. Якщо всі проекти є неможливими, найкращим є проект, який найближчий до того, щоб бути можливим, незалежно від значення цільової функції.

Загальний опис методів оптимізації в ANSYS

Методи оптимізації виконують мінімізацію цільової функції. У програмі доступні два методи: метод апроксимації (subproblem approximation method) і метод першого порядку (first order method). Метод апроксимації – це метод нульового порядку, що забезпечує ефективне розв'язання більшості конструкторських задач. Метод першого порядку ґрунтується на оцінці чутливості проекту до зміни певних чинників і більше підходить для розв'язання задач, які вимагають високої точності [7].

Крім того, користувач може застосувати інший оптимізаційний алгоритм, в цьому випадку алгоритм ANSYS буде пропущений.

Як при використанні методу апроксимації, так і методу першого порядку програма виконує серію ітерацій. Протягом кожної ітерації виконується розрахунок початкового

проекту, оцінюється відповідність результатів розрахунку певним критеріям якості і за необхідності здійснюється зміна проекту. Цей процес продовжується до того часу, поки не будуть виконані певні умови.

Змінні стану і обмеження на змінні проекту використовуються, щоб обмежити зміни в проекті, і приводять до оптимізаційної задачі з обмеженнями. ANSYS перетворює цю задачу в оптимізаційну задачу без обмежень. При проведенні оптимізації за кожним з методів програма враховує обмеження, накладені на змінні стану, додаючи до цільової функції штрафні функції.

Метод апроксимації

При використанні методу апроксимації програма на кожній ітерації виконує апроксимацію цільової функції і змінних стану (методом найменших квадратів) квадратичними функціями змінних проекту [7].

Цільова функція в комплексній формі апроксимації може бути записана в квадратичному вигляді з перехресними членами:

$$\hat{f} = a_0 + \sum_i^n a_i x_i + \sum_i^n \sum_j^n b_{ij} x_i x_j . \quad (97)$$

Для апроксимації використовуються значення цільової функції і змінних стану на попередніх ітераціях (тобто для попередніх наборів параметрів).

Необхідно зазначити, що в документації до ANSYS не описані деякі деталі алгоритму побудови даних апроксимацій (обчислення вагових коефіцієнтів для різних наборів параметрів). Крім того, існують обмеження на кількість наборів параметрів, які одночасно враховуються при апроксимації.

Норма найменшої зваженої квадратичної помилки для цільової функції має вигляд

$$E^2 = \sum_{j=1}^{n_d} \phi^{(j)} (f^{(j)} - \hat{f}^{(j)})^2 , \quad (98)$$

де $\phi^{(j)}$ - ваговий коефіцієнт, пов'язаний з набором параметрів; n_d - поточне число наборів параметрів.

Після побудови апроксимацій програма перетворює оптимізаційну задачу з обмеженнями в задачу без обмежень, за допомогою алгоритму SUMT знаходить екстремум апроксимації цільової функції і призначає на наступній ітерації значення змінних проекту, що відповідають цьому екстремуму. Ця процедура повторюється і на наступних ітераціях.

Штрафна функція, яку потрібно мінімізувати:

$$F(x, p_k) = \hat{f} + f_0 p_k \left(\sum_{i=1}^n X(x_i) + \sum_{i=1}^{m_1} G(\hat{g}_i) + \sum_{i=1}^{m_2} H(\hat{h}_i) + \sum_{i=1}^{m_3} W(\hat{w}_i) \right) \quad (99)$$

X - штрафна функція змінних проекту і G, H, W - штрафні функції змінних стану.

Для прикладу біля верхньої межі штрафна функція змінних проекту записується у вигляді

$$X(x_i) = \begin{cases} c_1 + c_2 / (\bar{x} - x_i), & \text{якщо } x_i < \bar{x} - \varepsilon(\bar{x} - \underline{x}) \\ c_3 + c_4 / (\bar{x} - x_i), & \text{якщо } x_i \geq \bar{x} - \varepsilon(\bar{x} - \underline{x}) \end{cases} \quad (i = 1, 2, 3, \dots, n) \quad (100)$$

де c_1, c_2, c_3, c_4 - обмеження; ε - дуже мале додатне число.

Для змінних стану штрафна функція (рис. 26)

$$W(w_i) = \begin{cases} d_1 + d_2 / (\bar{w} - \hat{w}_i) & \text{if } \hat{w}_i < \bar{w}_i - \varepsilon(\bar{w}_i - \underline{w}_i) \\ d_3 + d_4(\hat{w} - \bar{w}) & \text{if } \hat{w}_i \geq \bar{w}_i - \varepsilon(\bar{w}_i - \underline{w}_i) \end{cases} \quad (i = 1, 2, 3, \dots, m_1) \quad (101)$$

де d_1, d_2, d_3, d_4 - обмеження.

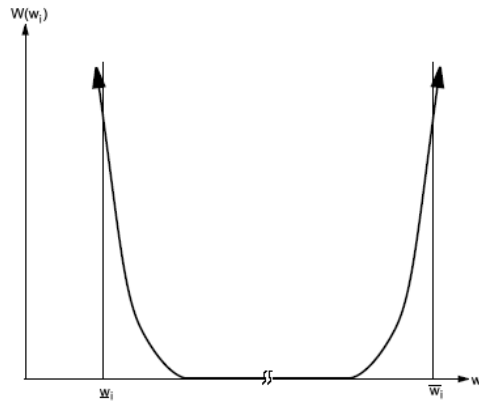


Рисунок 26 - Розширена внутрішня штрафна функція в просторі змінних стану

Користувач може вибирати тип апроксимуючої функції. Можна використовувати лінійну функцію, квадратичну функцію без перехресних членів і квадратичну функцію з перехресними членами.

Для початку ітерацій за методом апроксимації необхідна наявність певної кількості наборів параметрів (для побудови апроксимуючої функції). У разі їх відсутності програма створить їх сама, випадковим чином варіюючи змінні проекту усередині їх меж.

Оскільки це випадкові набори параметрів, то збіг може бути повільним. Іноді можна прискорити збіг, створивши декілька можливих наборів параметрів. Це може бути зроблено шляхом створення декількох випадкових наборів параметрів і виключення всіх неможливих наборів параметрів. Крім того, можна створити початкові набори параметрів шляхом виконання декількох одиночних циклів аналізу, зазначаючи нові значення змінних проекту перед кожним циклом.

У кінці кожного циклу аналізу виконується перевірка збігу і умов переривання оптимізації. Задача вважається тією, що виконана, якщо поточний, попередній і найкращий проекти (набори параметрів) є можливими і виконується одна з таких умов:

– різниця значень цільової функції між кращим можливим проектом і поточним проектом менше похибки збігу цільової функції

$$|f^{(j)} - f^{(j-1)}| \leq \tau; \quad (102)$$

– різниця значень цільової функції між двома останніми проектами менше похибки збігу цільової функції

$$|f^{(j)} - f^{(b)}| \leq \tau; \quad (103)$$

– різниці значень всіх змінних проекту між кращим можливим проектом і поточним проектом менше їх похибки збігу

$$|x_i^{(j)} - x_i^{(j-1)}| \leq \rho_i \quad (i = 1, 2, 3, \dots, n); \quad (104)$$

– різниці значень всіх змінних проекту між двома останніми проектами менше їх похибки збігу

$$|x_i^{(j)} - x_i^{(b)}| \leq \rho_i \quad (i = 1, 2, 3, \dots, n), \quad (105)$$

де τ і ρ_i - похибки збігу цільової функції і змінних проекту.

Користувач може зазначити похибки збігу цільової функції і змінних проекту як TOLER в команді **OPVAR**.

Іноді процедура оптимізації може бути перервана до досягнення збігу. Це відбувається у разі виконання однієї з наведених нижче умов:

- виконана названа кількість ітерацій;
- кількість послідовних неможливих проектів досягла вказаної межі.

Збіг не завжди означає знаходження глобального мінімуму. Він означає тільки те, що був виконаний один із названих вище критеріїв. Тому саме користувач повинен визначити, чи був проект достатньо оптимізований. Якщо це не так, то можна виконати додаткові ітерації.

Метод першого порядку

Метод першого порядку використовує інформацію про похідні залежних змінних щодо змінних проекту. Цей метод дуже точний і добре розв'язує задачі з великими діапазонами зміни залежних змінних, проте вимагає великих обчислювальних ресурсів [7].

Задача формулюється в такому вигляді:

$$Q(x, q) = \frac{f}{f_0} + \sum_{i=1}^n P_x(x_i) \left(\sum_{i=1}^{m_1} P_g(g_i) + \sum_{i=1}^{m_2} P_h(h_i) + \sum_{i=1}^{m_3} P_w(w_i) \right), \quad (106)$$

де Q - безрозмірна цільова функція; P_x, P_g, P_h, P_w - штрафи, які визначаються конструкцією і змінними стану; f_0 - початкова цільова функція.

Для прикладу для змінної стану, яка обмежена верхньою межею, штрафна функція запишеться як

$$P_g(g_i) = \left(\frac{g_i}{g_i + \alpha_i} \right)^{2\lambda}, \quad (107)$$

де λ - ціле число, яке стає дуже великим, коли виходить за межі, і, навпаки, дуже маленьким, коли знаходиться всередині діапазону.

При використанні методу першого порядку програма перетворює оптимізаційну задачу з обмеженнями в задачу без обмежень, а потім на кожній ітерації обчислює градієнт цільової функції за змінними проекту. Для обчислення кожної частинної похідної програма присвоює невеликий приріст відповідної змінної проекту, залишаючи значення інших змінних проекту колишніми (метод покоординатного спуску), і виконує розрахунок конструкції з даним набором параметрів.

Після обчислення всіх частинних похідних визначається напрям пошуку екстремуму на даній ітерації. Необхідно зазначити, що в загальному випадку пошук здійснюється не у напрямку градієнта, для визначення напрямку пошуку використовується складніша залежність. Потім здійснюється лінійний пошук екстремуму цільової функції за даним напрямком.

Для кожної ітерації оптимізації (j) шукається вектор напрямку пошуку екстремуму $d^{(j)}$. Наступна ітерація знаходиться з такого рівняння:

$$x^{(j+1)} = x^{(j)} + s_j d^{(j)}. \quad (108)$$

Одержаний з $x^{(j)}$ лінійний параметр пошуку s_j відповідає мінімальній величині Q в напрямку $d^{(j)}$. Для знаходження розв'язання для s_j використовується комбінація методів золотого перетину і локального квадратичного методу підбору. Діапазон s_j є обмеженим:

$$0 \leq s_j \leq \frac{S_{\max}}{100} s_j^* \quad (109)$$

де s_j^* - найбільший можливий розмір кроку для лінійного пошуку в поточній ітерації (розраховується програмою); S_{\max} - максимальний (у відсотках) розмір кроку лінійного пошуку.

Користувач може вказати прирости змінних проекту, які використовуються для обчислення градієнта, а також граничне значення кроку лінійного пошуку (ввести SIZE в команді **OPFRST**).

Розв'язання задачі безумовної мінімізації (106) ґрунтується на послідовній генерації напрямків пошуку і на внутрішніх коригуваннях поведінки поверхневого параметра q . Для початкової ітерації $j = 0$ напрямком пошуку передбачено буде негативним стосовно градієнта незв'язаної цільової функції. Для початкової ітерації як метод пошуку використовується метод найшвидшого спуску. Для подальших ітерацій $j > 0$ зв'язані напрямки формуються відповідно до рекурсивної формули Полака-Рібера

$$d^{(j)} = -\nabla Q(x^{(j)}, q_k) + r_{j-1} d^{j-1}$$

$$r_{j-1} = \frac{[\nabla Q(x^{(j)}, q) - \nabla Q(x^{(j-1)}, q)]^T \nabla Q(x^{(j)}, q)}{|\nabla Q(x^{(j-1)}, q)|^2} \quad (110)$$

Вектор градієнта обчислюється за допомогою апроксимації у такому вигляді:

$$\frac{\partial Q(x^{(j)})}{\partial x_i} = \frac{Q(x^{(j)} + \Delta x e) - Q(x^{(j)})}{\Delta x_i} \quad (111)$$

де e - вектор, який дорівнює або 1, або 0; $\Delta x_i = \frac{\Delta D}{100} (\bar{x}_i - \underline{x}_i)$, ΔD - правостороння різниця (у відсотках) розміру кроку (ввести DELTA в команді **OPFRST**).

Таким чином, кожна ітерація розділяється на набір субітерацій, який включає пошук напрямку і обчислення градієнта. У зв'язку з цим одна оптимізаційна ітерація для методу першого порядку включає декілька циклів аналізу.

Знайдений таким чином екстремум використовується як початкова точка для наступної ітерації і т.д.

Ітерації продовжуються до того часу, поки не буде досягнута збіжність або умови переривання процесу оптимізації. Задача вважається такою, що збіглася, якщо поточний, попередній і найкращий проекти (набори параметрів) такі, що виконується одна з наступних умов:

- різниця значень цільової функції між кращим проектом і поточним проектом менше похибки збігу цільової функції

$$|f^{(j)} - f^{(j-1)}| \leq \tau; \quad (112)$$

- різниця значень цільової функції між попереднім проектом і поточним проектом менше похибки збігу цільової функції

$$|f^{(j)} - f^{(b)}| \leq \tau. \quad (113)$$

Процедура оптимізації може бути перервана до досягнення збігу. Це відбувається у випадку, якщо виконана максимальна кількість ітерацій, зазначена користувачем.

У порівнянні з методом апроксимації метод першого порядку є точнішим. Проте висока точність методу першого порядку не завжди гарантує отримання найкращого розв'язання. Для методу першого порядку можливий збіг при неможливому наборі параметрів проекту. В цьому випадку швидше за все був знайдений локальний мінімум або не існує можливих наборів параметрів проекту.

Якщо це трапилося, може бути корисним проведення оптимізації методом апроксимації, оскільки це більш добрий засіб для дослідження всієї області варіювання параметрів проекту. Крім того, буде корисним одержати випадкові набори параметрів проекту, щоб знайти область їх можливих значень (якщо вона існує), а потім перезапустити метод першого порядку, використовуючи можливий набір параметрів проекту як початкову точку.

Засоби оптимізації в ANSYS

Окрім двох методів оптимізації, в програмі ANSYS доступні п'ять різних засобів оптимізації [7].

Засоби оптимізації використовуються для оцінки і розуміння області варіювання параметрів проекту. Вони забезпечують не оптимізацію цільової функції, а автоматичне отримання декількох наборів параметрів проекту при певному законі зміни змінних проекту. Для використання цих засобів не потрібна наявність цільової функції, проте змінні проекту повинні бути визначені.

Засоби оптимізації, доступні в ANSYS:

- одноразовий аналіз (single-loop analysis tool) полягає у виконанні розрахунку одного варіанта конструкції зі встановленими у даний момент змінними проекту;
- випадкове варіювання (random tool) виконує вказане число циклів аналізу, встановлюючи кожного разу для змінних проекту випадкові значення;
- сканування області варіювання параметрів (sweep tool) створює задану кількість наборів параметрів, по черзі варіюючи кожен змінну проекту в початковому наборі параметрів через весь діапазон її зміни. Значення інших змінних проекту при цьому залишаються незмінними;
- аналіз (factorial tool) чинника створює набори параметрів, аналізуючи конструкцію при різних поєднаннях крайніх значень змінних проекту;
- градієнтний аналіз (gradient tool) обчислює градієнти цільової функції і змінних стану стосовно змінних проекту.