

YAPAY ARI KOLONİ ALGORİTMASI İLE TAŞIT SALINCAK KOLONUN OPTİMUM BOYUTLARININ BULUNMASI

Ali Rıza YILDIZ¹

¹aliriza.yildiz@btu.edu.tr Bursa Teknik Üniversitesi, Makine Mühendisliği Bölümü, 16200 Bursa

ÖZET

Ürün geliştirme süreci ürün maliyeti, kalitesi, imalat işlemleri, kullanım ve bakım maliyeti üzerinde önemli etkiye sahiptir. Bu nedenle, ürün tasarım sürecinde, optimum tasarımın elde edilmesi gerekmektedir. Bu çalışmada otomobillerin ön süspansiyon sisteminde kullanılan salıncak kolunun, sırasıyla topoloji ve şekil optimizasyonu yapılarak optimum boyutları bulunmuştur. Şekil optimizasyonu için yapay arı koloni algoritması kullanılmıştır.

Anahtar Sözcükler: Topoloji optimizasyonu, Şekil Optimizasyonu, Taşıt Salıncak Kolu, Yapay Arı Koloni Algoritması

ABSTRACT

In this research, a vehicle swing arm is optimized using topology and shape optimization approaches. Artificial bee colony algorithm is used for shape design optimization.

Keywords: Topology optimization, Shape optimization, Vehicle design, Artificial bee colony algorithm

1. GİRİŞ

Ürün geliştirme süreci, ürün ömrü içinde maliyet açısından en etkili olan aşamadır. Ürünün toplam yaşam çevrimi maliyetinin yarıdan fazlasının tasarım aşaması sırasında belirlendiği düşünülürse, etkin imalat açısından amaçlanan nokta, tasarım aşamasında istenen kriterleri sağlayan ürünlerin tasarlanmasıdır. Bu nedenle alternatif tasarımlar içinde optimum olanın seçilmesi üretim açısından çok önem taşımaktadır.

Sayısal ve türev temelli optimizasyon yöntemlerine alternatif olarak genetik algoritmalar, parçacık sürüsü optimizasyon algoritması, armoni arama algoritması, farksal gelişim algoritması, yapay bağıklık algoritması gibi değişik popülasyon temelli optimizasyon algoritmaları geliştirilmiş ve farklı alanlardaki optimizasyon çalışmalarında yaygın bir şekilde kullanılmıştır [1-15].

Bu araştırmanın amacı, ürün tasarım sürelerini azaltmak ve istenen kriterleri sağlayan optimum ürünleri kısa sürede tasarlamaya katkı koyacak optimizasyon yöntemlerini ortaya koymak ve optimum ürün tasarımında etkin bir şekilde kullanılabileceğini göstermektir. Bu kapsamda otomobillerin ön süspansiyon sistemlerinde kullanılan salıncak kolunun sırasıyla topoloji ve şekil optimizasyonu yapılmıştır. Şekil optimizasyonu için yapay arı koloni algoritması kullanılmıştır.

2. YAPAY ARI KOLONİ ALGORİTMASI

Yapay arı koloni algoritması (YAK), bal arısı sürülerinin kendilerine özgü zeki davranışlarını örnek alan, arıların besin ararken kullandıkları yöntemlerden esinlenerek geliştirilmiş popülasyon temelli bir optimizasyon yöntemidir. YAK ilk olarak Karaboğa [16] tarafından geliştirilmiş ve algoritma kompozit yapıların optimizasyonu, tornalama problemlerinin optimizasyonu, hidrolik mühendisliğinden havacılık ve uzay bilimi gibi farklı bilim dallarında başarıyla uygulanmıştır [17-19]. Doğal yaşamda bal toplayan bir arı kolonisi içinde görev paylaşımı vardır. Kolonide işçi, gözlemci ve izci arılar farklı görevleri yerine getirmektedirler. İşçi arıların görevi, adresi verilmiş bal kaynaklarına gidip, çiçek özlerini toplamak ve topladığı balları kovana boşaltmaktır. Ancak işçi arıların başka bir görevi daha vardır. Bal topladığı kaynağın kalitesini gözlemci arılara anlatması gerekmektedir. Gözlemci arıların görevi ise yeni bal kaynakları tespit edip, işçi arıları bu kaynaklara yönlendirmektir. Yeni kaynakların belirlenebilmesi için işçi arıların daha önceki gözlemleri dinlenerek analiz edilmektedir. Diğer yandan izci arılar rastgele olarak çevreyi dolaşmakta ve sadece kendi gözlemlerini dikkate alarak yeni bal kaynakları aramaktadırlar. Algoritmada nektar kaynağının adresinin belirlenmesi süreci, optimizasyon problemine bir çözüm olarak sunulmaktadır. Nektar kaynağının büyüklüğü ise ilgili çözümdeki uygunluk derecesi olarak alınmaktadır. Algoritmaya göre her bir bal kaynağı sadece bir işçi arı tarafından toplanmaktadır. Bu nedenle, popülasyondaki çözüm sayısı, işçi arı sayısı ile aynı olmaktadır [16-19]. Algoritmanın işleyişi aşağıda belirtilen beş temel adımda gerçekleştirilmektedir.

1. Bal kaynağı bölgelerinin ilk değerlerinin belirlenmesi,
 2. İşçi arıların belirlenen bal kaynaklarına yönlendirilmesi,
 3. Bal kaynağı olmaya aday bölgeler için olasılık hesaplamalarının yapılması,
 4. İşçi arıların aktardıklarına bağlı olarak, gözlemci arıların yeni bal kaynaklarını belirlemesi,
 5. Mevcut bal kaynaklarının kullanım dışı bırakılma kararının verilmesi şeklindedir.
- Algoritmada bal kaynakları bölgelerine ait ilk değerler denklem 1 kullanılarak atanmaktadır [16-19].

$$x_{ij} = x_j^{\min} + rand(0,1)(x_j^{\max} - x_j^{\min}) \quad (1)$$

Denklemden j üretilen kaynak sayısını, i ise optimizasyon parametre sayısını temsil eder. İkinci adımda işçi arı belirlenmiş olan bal kaynağından nektar toplamaktadır. Aynı zamanda bulunduğu kaynağın yakınında bulunan başka bir bal kaynağına ilişkin bir analiz yaparak, yeni kaynağın kalitesini değerlendirmektedir. ϕ , $[-1, 1]$ aralığında rastgele değer alan reel sayı olmak üzere yeni bal kaynağı araştırması v_{ij} hakkında yaptığı hesaplama denklem 2 de verilmiştir.

$$v_{ij} = x_{ij} + \phi_{ij}(x_{ij} - x_{kj}) \quad (2)$$

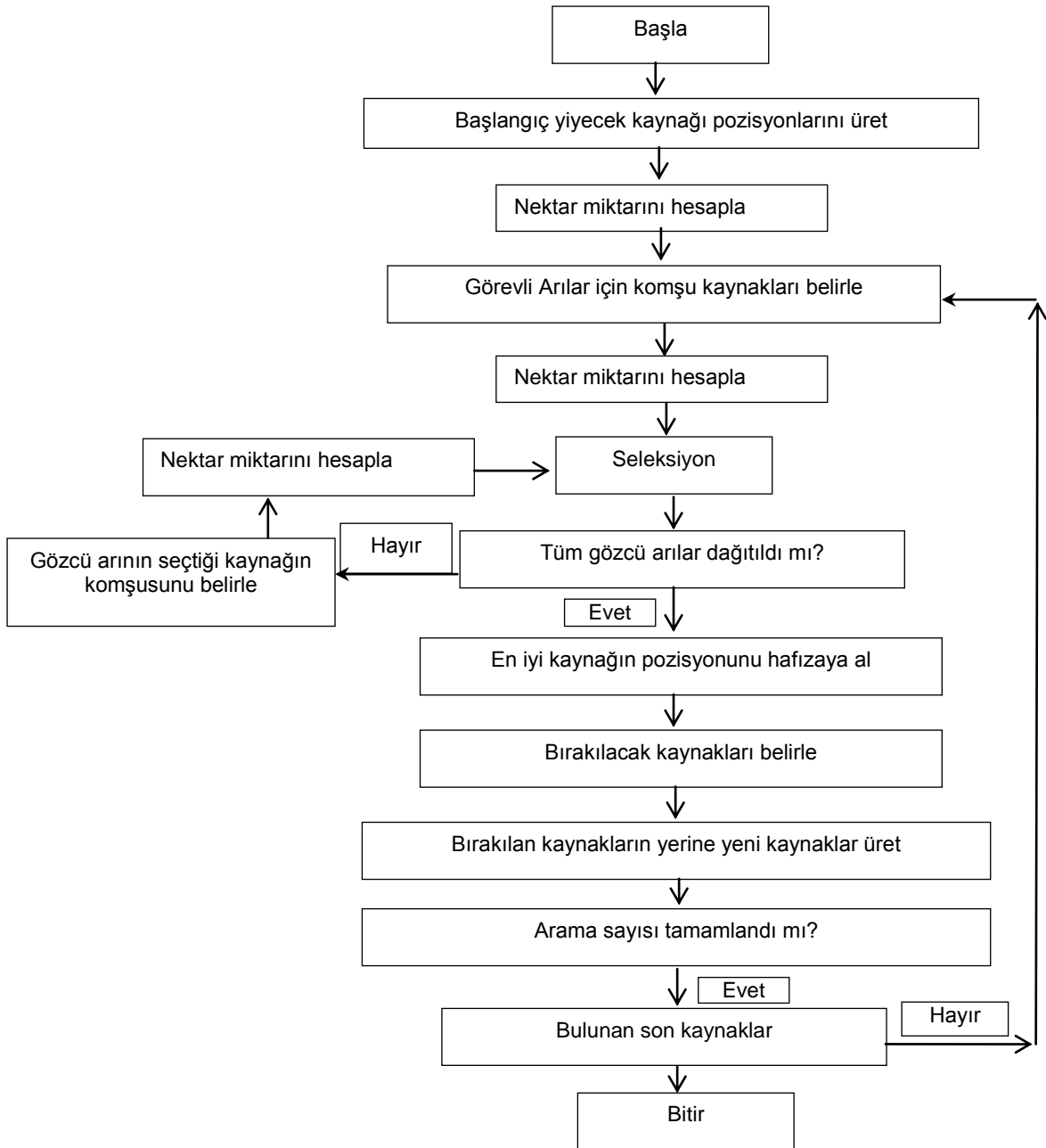
Değerlendirme yapılırken v_{ij} 'nin daha önceden belirlenen parametre sınırları dışına çıkması durumunda denklem 3 e göre öteleme yapılır ve uygunluk değerleri denklem 4 e göre hesaplanır [16-19].

$$v_{ij} = \begin{cases} x_j^{\min}, & v_{ij} < x_j^{\min} \\ v_{ij}, & x_j^{\min} \leq v_{ij} \leq x_j^{\max} \\ x_j^{\max}, & v_{ij} > x_j^{\max} \end{cases} \quad (3)$$

$$fitness_i = \begin{cases} 1/(1+f_i), & f_i \geq 0 \\ 1 + abs(f_i), & f_i < 0 \end{cases} \quad (4)$$

Kovan içinde uygunluk oranları gözlemci arılar tarafından değerlendirilir ve nektar miktarlarına göre seçim yapılır. Bu çalışmada YAK algoritmasında, seçim yapmak için rulet çemberi yöntemi kullanılmıştır.

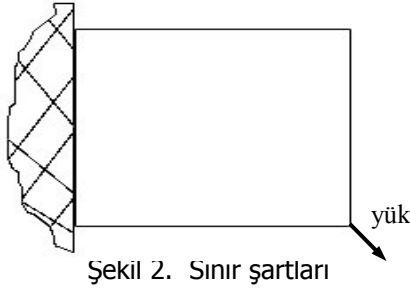
Algoritmanın son kısmı, nektarı tükenmiş bal kaynağının belirlenmesidir. İşçi ve gözcü arıların arama sürecinin tamamlanması ile oluşan bir döngü sonucunda, algoritma tükenmiş bir bal kaynağını belirleyebilir. Bu amaçla, arama sürecinde güncellenen sayıcılar kullanılır. Eğer herhangi bir bal kaynağına ait sayıcı algoritma tarafından belirlenen ve limit olarak bilinen kontrol parametresinden daha büyükse, bahsi geçen kaynak tükenmiş olarak değerlendirilir ve terk edilir. Terk edilen alana ait işçi arı algoritmadan çıkarılır. Yeni kaynağı bulan izci arı ise işçi arıya dönüşür ve bulunduğu kaynaktan bal toplamaya başlar ve o kaynak için limit değeri sıfırlanır. YAK algoritmasında her çevrimde sadece bir kâşif arının çıkmasına izin verilir. Algoritmanın sonlandırılması iki şekilde olabilir. Bu koşullar maksimum sayıda döngünün gerçekleşmesi ya da belirlenen bir hata tolerans değerinin aşılması şeklinde kullanıcı tarafından belirlenir. Yapay arı koloni algoritmasının akış diyagramı Şekil 1’de gösterilmiştir.



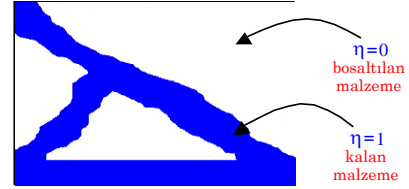
Şekil 1. Yapay arı koloni algoritmasının akış diyagramı

3. BİLGİSAYAR DESTEKLİ OPTİMİZASYON

Bu çalışmada topoloji optimizasyonu ve analizler için ANSYS programı kullanılmıştır [20]. Topoloji optimizasyonu ile ilgili örnek bir uygulama şekil 2' de verilmiştir. Bu örnekte, amaç fonksiyonu olarak kompliansın minimizasyonu, kısıtlayıcı olarakta hacmin % 60 azalması seçilmiştir. Şekil 3 de görüldüğü gibi topoloji optimizasyonu sonucunda $\eta=0$ ile gösterilen bölge boşaltılması önerilen bölgeyi, $\eta=1$ ile gösterilen bölgede optimizasyon sonucu kalacak bölgeyi göstermektedir [21-22]. .



Şekil 2. Sınır şartları



Şekil 3. Topoloji optimizasyonu sonrası şekil

Malzeme dağılımını esas alan topoloji optimizasyonu işlem adımları aşağıda verilmiştir.

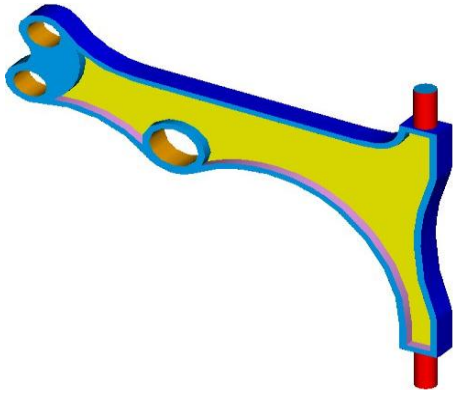
- Adım1: Parça modelinin oluşturulması
- Adım2: Eleman tipinin seçilmesi
- Adım3: Optimizasyon yapılacak ve yapılmayacak kısımların belirlenmesi.
- Adım4: Sınır şartlarının uygulanması
- Adım5: Optimizasyon işleminin kontrolü
- Adım6: Sonuçların görüntülenmesi,

4. ÖRNEK ÇALIŞMA

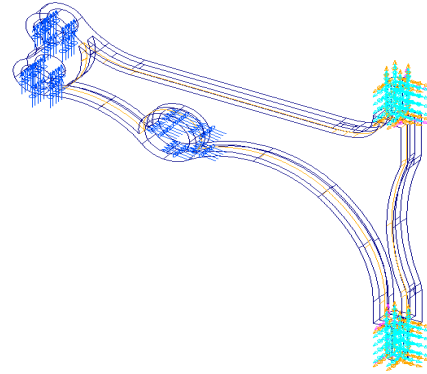
Bu çalışmada örnek uygulama parçası olarak taşıt salıncak kolu alınmıştır. Optimizasyon işlem takip eden alt bölümlerde anlatılmıştır.

4.1 Salıncak Kolunun Topoloji Optimizasyonu

Taşıtlarda kullanılan salıncak kolunun topoloji optimizasyonu çalışmasına, şekil 4 de görülen parçanın taslak modelinin oluşturulması ile başlamıştır. Şekil 5' de mevcut sınır şartları görülmektedir.

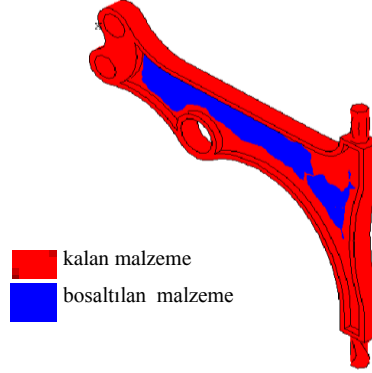


Şekil 4. Salıncak kolu başlangıç modeli



Şekil 5. Sınır şartları

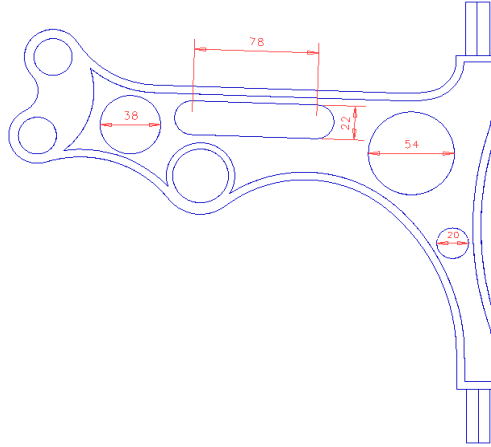
Başlangıç modeli oluşturulan parça şekil 5 de verilen sınır şartlarında amaç fonksiyonu olarak komplians minimizasyonu, kısıtlayıcı olarak % 60 hacim azalması seçilerek topoloji optimizasyonu yapılmış ve şekil 6 da görülen malzeme dağılımı elde edilmiştir.



Şekil 6. Topology optimizasyonu sonrası malzeme dağılımı

4.2 Salıncak Kolunun Şekil Optimizasyonu

Şekil 6 da elde edilen dağılımı göre salıncak kolu şekil 7 de görüldüğü gibi tekrar tasarlanmıştır. Tasarım sonrası elde edilen 5 parametre dikkate alınarak salıncak kolunun yapay arı koloni algoritması ile şekil optimizasyonu yapılmıştır.



Şekil 7. Tasarım değişkenleri

Salıncak kolu için optimizasyon öncesi ve sonrası elde edilen sonuçlar Tablo 1'de karşılaştırılmalı olarak verilmiştir.

Tablo 1. Salıncak kolu optimizasyonu için karşılaştırılmalı sonuçlar

	Hacim (mm ³)	Maks. gerilme (N/mm ²)
Başlangıç	212619	540.9
Topoloji Optimizasyonu Sonrası	184887	685.0
Şekil Optimizasyonu Sonrası	183423	782.0
Parçacık sürüsü optimizasyonu algoritması	179680	788.0
Yapay arı koloni algoritması	175800	795.1

5. SONUÇLAR

Son yıllarda bilgisayar destekli tasarım analiz ve optimizasyon yöntemleri ürün geliştirme zamanını ve maliyetleri azaltabilmek için otomotiv sektöründe yaygın olarak kullanılmaya başlanmıştır. Bu çalışmanın sonuçlarında görüleceği üzere, yapısal optimizasyon tekniklerinin kullanılması ürün geliştirme süreçlerini ve ürün maliyetlerini minimize etmek için çok önemlidir. Bu çalışmada taşıtlarda kullanılan salıncak kolunun sırasıyla topoloji optimizasyonu ve yapay arı koloni algoritması kullanılarak şekil optimizasyonu yapılmış ve optimum tasarım elde edilmiştir. Bu çalışmada elde edilen sonuçlar, topoloji optimizasyonu ve yapay arı koloni algoritmasının taşıt elemanlarının optimum tasarımında etkin bir şekilde kullanılabileceğini göstermektedir.

6. KAYNAKLAR

- [1] YILDIZ, A.R., SOLANKI, K., Multi-objective optimization of vehicle crashworthiness using a new particle swarm based approach, **International Journal of Advanced Manufacturing Technology**, 59(1-4): 367-376, (2012).
- [2] YILDIZ, A.R., SAITOU, K., Topology Synthesis of Multi-Component Structural Assemblies in Continuum Domains, Transactions of ASME, **Journal of Mechanical Design** 133(1) 011008-1– 011008-9, (2011).
- [3] BUREERAT, S., LIMTRAGOOL, J., Performance enhancement of evolutionary search for structural topology optimisation, **Finite Elements in Analysis and Design**, 42(6), pp.547-566, (2006).
- [4] RENNERRA, G., EKA'R, A., Genetic algorithms in computer aided design, **Computer-Aided Design** 35, pp.709–726, (2003).
- [5] YILDIZ, A.R., A new hybrid particle swarm optimization approach for structural design optimization in the automotive industry, Proceedings of the Institution of Mechanical Engineers, Part D: **Journal of Automobile Engineering**, DOI: 10.1177/0954407012443636, (2012).
- [6] FERHAT, E., ERKAN, D., SAKA, M.P., Optimum design of cellular beams using harmony search and particle swarm optimizers, **Journal of Constructional Steel Research** 67(2) pp.237-247, (2011).
- [7] OMKAR, S.N., SENTHILNATH, J., KHANDELWAL, R., NARAYANA NAIK, G., GOPALAKRISHNAN, S., Artificial Bee Colony (ABC) for multi-objective design optimization of composite structures, **Applied Soft Computing** 11, pp.489–499, (2011).

- [8] CAMP, C.V., BICHON, B.J., STOVALL, S.P., Design of steel frames using ant colony optimization, **Asce-Journal of Structural Engineering** 131, pp.367-525, (2005).
- [9] SAKA, M.P., Optimum design of steel grillage systems using genetic algorithms, **Computer Aided Civil Infrastruct Eng** 13, pp.233-238, (1998).
- [10] OMKAR, S.N., SENTHILNATH, J., KHANDELWAL, R., NARAYANA NAIK, G., GOPALAKRISHNAN, S., Artificial immune system for multi-objective design optimization of composite structures, **Engineering Applications of Artificial Intelligence** 21, pp.1416-1429, (2008).
- [11] Hybrid immune-simulated annealing algorithm for optimal design and manufacturing, **International Journal of Materials and Product Technology**, 34(3), 217-226, (2009).
- [12] YILDIZ, A.R., A new design optimization framework based on immune algorithm and Taguchi method, **Computers in Industry**, 60(8), 613-620, (2009).
- [13] YILDIZ, A.R., Hybrid Taguchi-Harmony Search Algorithm for Solving Engineering Optimization Problems, **International Journal of Industrial Engineering Theory, Applications and Practice**, 15(3), 286-293, (2008).
- [14] YILDIZ, A.R., A novel particle swarm optimization approach for product design and manufacturing, **International Journal of Advance Manufacturing Technology** 40, pp.617-628, (2009).
- [15] YILDIZ, A.R., A novel hybrid immune algorithm for global optimization in design and manufacturing, **Robotic and Computer Integrated Manufacturing** 25, pp.261-270, (2009).
- [16] KARABOGA, D., An idea based on honey bee swarm for numerical optimization, Technical report TR06. **Computer Engineering Department**, Erciyes University, Turkey, (2005).
- [17] KARABOGA, D., OZTURK, C., A novel clustering approach: Artificial Bee Colony (ABC) algorithm, **Applied Soft Computing**, 11(1), 652-657, (2011).
- [18] KARABOGA, D., Basturk, B., A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm, **Journal of Global Optimization**, 39, 459-471, (2003).
- [19] YILDIZ, A.R., A new hybrid artificial bee colony algorithm for robust optimal design and manufacturing, **Applied Soft Computing**, <http://dx.doi.org/10.1016/j.asoc.2012.04.013>, (2012).
- [20] ANSYS 12, Help, Swanson Analysis Systems Inc., USA.
- [21] BENDSOE, M.P., KIKUCHI, N., Generating optimal topologies in structural design using a homogenization method, **Comput. Meth. Appl. Mech. Engineering**, 71, 197-224, (1988).
- [22] YANG, R.J., CHUANG, C.H., Optimal topology design using linear programming, **Structural Optimization**, Vol.68, p.265-290, (1993).