We are IntechOpen, the world’s leading publisher of Open Access books
Built by scientists, for scientists

4,500
Open access books available

118,000
International authors and editors

130M
Downloads

154
Countries delivered to

TOP 1%
Our authors are among the most cited scientists

12.2%
Contributors from top 500 universities

WEB OF SCIENCE™
Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com
1. Introduction

Obstacle avoidance methods for mobile robots have proposed in a broad range of studies and the availabilities have been discussed. Most of these studies regard the robots as points or circles and control methods of the translational movements are discussed. In these studies, it is pointed out that a non-circle robot can be transformed into a point robot by expanding the obstacles by the largest radius or maximum size of the robot. The effectiveness of avoiding obstacles by these approaches have been confirmed, however, according to the shape of the robot, these approaches reduce and waste the available free-space and can decrease the likelihood of getting to the goal. If wide-robots, which are horizontally long, are regarded as circles according as conventional approaches, they have possibilities not to go through between two divided objects due to the largest radius of the robot, even if they ought to be able to go through by using their shortest radius. This suggests necessity of suitable orientation angle at the moment of avoidance. Consequently, to enable wide-robots to avoid obstacles safely and efficiently, it is necessary to control not only the translational movement but also the rotational movement. In our current research, wide-robots with omni-directional platform have been employed, as shown in Fig.1. In situations like Fig.1, both wide-robots can go through only by changing the orientation angle in real time.

Some researches focus attention on the orientation angle of the robot (Kavraki, 1995)(Wang & Chirikjian, 2000). In these studies, by convolving the robot and the obstacle at every orientation and constructing the C-space, the suitable orientation angles of the robot for path planning are decided. However, these methods need environmental map and do not show the effectiveness for autonomous mobile robots about avoidance of unknown obstacles in these studies. Therefore, in order to avoid unknown obstacles reactively considering the orientation angle, the wide-robot needs an algorithm that can decide the orientation angle and rotational velocity command on the spot based on the current obstacle information.

Meanwhile, decision methods of the translational movement have been proposed in many studies (Wang et al., 2000) (Du et al. 2007) (Khatib, 1986) (Borenstein & Koren, 1991) (Dieter, 1997), we employ fuzzy potential method (FPM) (Tsuzaki & Yoshida, 2003). This method realizes some tasks in dynamic environment by fuzzy calculation about desire for each direction of the robot. In this research, it was shown that wheeled robots succeeded getting to the goal with conveying a soccer ball and avoiding obstacles.
In this paper, a control method using a capsule-shaped case is described for both translational and rotational movement based on the FPM, and it takes into consideration the width of the robot. With this new approach, real-time control of the orientation angle is easily achieved. Conventional FPM has only been able to deal with translational velocity. The proposed method is able to control the rotational and translational velocity simultaneously within the framework of FPM.

2. Capsule case

2.1 Need to consider the width of mobile robot

In recent years, non-circle robots have been developed, of which vertically long robots, wide-robots, appear. In studies of humanoid robots, the robots have two arms mounted to the stationary torso with wheels because these robots can be used in terms of mobility, manipulation, whole-body activities, and human-robot interaction (Ambrose et al., 2004) (Du et al., 2007). During last two decades, vast number of algorithms of obstacle avoidance for mobile robot, and recently some researches and developments of the mobile robot in practical use have been reported. These robots have problems that conventional methods are inconvenient for applying for the wide-robot because most of conventional methods of obstacle avoidance regard the robot as points or circles. Or due to the postulate of the conventional methods, the robots have needed to be designed in a circle. We also have developed the wide-robot, which has torso with two arms and a head, for making the robot perform not only moving but also communication with human by means of gestures or speeches based on a perspective of human interaction. This robot is also horizontally long. In addition, when the robot opens an arm slightly, as shown in Fig. 1, or both arms, it becomes increasingly harder to apply conventional methods. If these wide-robots are regarded as circles according as conventional approaches, they have possibilities not to go through between two divided objects due to the largest radius of the robot, even if they ought to be able to go through by using their shortest radius. In this study, enable the wide-robots, which move automatically, to move smoothly and safely in the environment with obstacles, a capsule case is introduced.

2.2 Design of capsule case

The capsule shaped case is modeled by two circles and two lines tangent to the circles as shown in Fig.2.
Fig. 2. A capsule case

This closed contour is defined as $l(\phi)$ with the origin at the point $P_O$ and as follows equation:

$$1(\phi) = \begin{cases} 
  C_a / \cos \phi & \text{if } 0 \leq \phi < \phi_1, \\
  -C_a / \cos \phi & \text{if } \phi_1 \leq \phi < 2\pi, \\
  \sqrt{X(\phi)^2 + Y(\phi)^2} & \text{if } \phi_2 \leq \phi < \phi_3, \\
  \end{cases}$$  

(1)

where $\phi$ is clockwise from the back direction of the robot. $\phi_i$ are respectively

$\phi_1 = \arctan(C_L/C_a), \phi_2 = \pi - \arctan(C_L/C_a), \phi_3 = \pi + \arctan(C_R/C_a), \phi_4 = 2\pi - \arctan(C_R/C_a)$.

$X(\phi)$ and $Y(\phi)$ are calculated as following equations:

$$X(\phi) = \begin{cases} 
  -C_L - \sqrt{C_L^2 - (C_L^2 - C_a^2)\left[1 + \tan^2\left(\frac{\pi}{2} - \phi\right)\right]} & \text{if } \phi_1 \leq \phi < \phi_2, \\
  C_R + \sqrt{C_R^2 - (C_R^2 - C_a^2)\left[1 + \tan^2\left(\frac{\pi}{2} - \phi\right)\right]} & \text{if } \phi_3 \leq \phi < \phi_4, \\
  \end{cases}$$  

(2)

$$Y(\phi) = X(\phi) \cdot \tan\left(\frac{\pi}{2} - \phi\right).$$  

(3)

In the proposed method, $C_L$, $C_R$, $C_a$ are decided in a way that wide-robot shape falls within the capsule case.

Fig. 3. Examples of PMFs
3. Fuzzy potential method (FPM) using capsule case

3.1 Concept
In the fuzzy potential method (FPM), a current command velocity vector that takes into consideration element actions is decided by fuzzy inference. Element actions are represented as potential membership functions (PMFs), and then they are integrated by means of fuzzy inference. The directions on the horizontal axis in Fig. 3 correspond to the directions, which are from \(-\pi\) to \(\pi\) radian measured clockwise from the front direction of the robot. Grades for the directions are represented on the vertical axis. The grades, direction, and configured maximum and minimum speeds, are used to calculate the current command velocity vector.

Previously, the FPM dealt with the problem of the translational motion control in the same way as other control methods for autonomous mobile robots. In this paper, it is shown that modifying the FPM enables it to deal with rotational motion control, which is achieved concurrently with translational motions, within the FPM framework. In the modified framework as shown in Fig. 4, PMFs for translational motions and rotational motions are designed respectively based not only on the environmental information but also on the robot’s own condition. Environmental information and the robot’s own condition are treated separately and divided into a translation problem and a rotational problem. Then the PMFs of each problem are independently integrated using fuzzy inference. Finally, translational and rotational velocities, which are calculated by defuzzification of mixed PMFs, are realized by an omni-directional drive system.

3.2 PMF for translational motions

3.2.1 PMF for obstacles
In order to enable a wide-robot to avoid obstacles safely and efficiently in real time, a concave shaped PMF \(\mu_{j}^{\theta}(j = 1, 2, \ldots, n)\), which is considering the capsule case, is generated. This PMF is specified by depth and width, which are calculated based on geometrical relation between an obstacle and a robot as shown in Fig. 6. By generating based on some variables, which are \(\varphi_{1}, \varphi_{2}, \varphi_{L}, \varphi_{R}, a\), and \(\varphi_{L, o}\) in Fig. 5, the choice of safe direction becomes possible. First, \(\varphi_{L}, \varphi_{R}, \varphi_{L}^{\prime}, \varphi_{R}^{\prime}\) are calculated as following equations:

Fig. 4. A concept of fuzzy potential method considering translational and rotational motion with an omni-directional platform

![Diagram](https://example.com/diagram.png)
Translational and Rotational Motion Control
Considering Width for Autonomous Mobile Robots Using Fuzzy Inference

Fig. 5. A wide-robot and an obstacle

Fig. 6. PMFs for translational motions: $\mu_{q_i}^L$ is a PMF for an obstacle (a), $\mu_{g}^L$ is a PMF for a goal (b)

\[
\varphi_L = \arccos \left( \frac{\|P_OQ_O\|^2 + \|P_1Q_O\|^2 - \|P_0P_1\|^2}{2\|P_OQ_O\|\|P_1Q_O\|} \right),
\]

\[
\varphi_R = \arccos \left( \frac{\|P_OQ_O\|^2 + \|P_0Q_O\|^2 - \|P_0P_R\|^2}{2\|P_OQ_O\|\|P_0Q_O\|} \right).
\]

\[
\varphi_L' = \begin{cases} 
\arcsin \left( \frac{D}{\|P_1Q_O\|} \right) & \text{if } D < \|P_1Q_O\|, \\
\pi - \arcsin \left( \frac{\|P_1Q_O\| - d_s}{D - d_s} \right) & \text{if } D \geq \|P_1Q_O\|.
\end{cases}
\]

www.intechopen.com
Next, as a measure to decide how far the robot should depart from the obstacle, \( a \) is defined as the depth of the concave shaped PMF. \( a \) is described as following equation:

\[
a = \begin{cases} 
\alpha - r_{r,o} & \text{if } r_{r,o} < \alpha \\
\alpha - \arcsin \left( \frac{D - d_s}{D - d_s} \right) & \text{if } D \geq P^\prime Q^\prime 
\end{cases}
\]

(8)

where \( r_{r,o} = (r_r, r_o) \) is current position vector of the obstacle relative to the robot.

If the current obstacle position is inside of a circle with radius \( \alpha \) from the robot position, the PMF for obstacle avoidance is generated. In other words, if a relative distance \( r_{r,o} \) is below \( \alpha \), \( a \) is defined and the concave shaped PMF corresponding to the obstacle is generated. \( D \) is decided to ensure the safety distance as following equation:

\[
D = C_a + r_o + d_s
\]

(9)

where \( C_a \) is the minimum length of capsule case as shown in Fig.2. \( r_o \) and \( d_s \) denote respectively the radius of the obstacle and safety distance. \( \varphi_{r,o} \) is the angle of direction to the obstacle relative to the robot, which is calculated as following equation:

\[
\varphi_{r,o} = \arctan \left( \frac{r_y}{r_x} \right)
\]

(10)

The PMF \( \mu^1_{oo} \) is generated for all obstacles which the robot has detected. And then, they are all integrated by calculating logical product \( \mu^1_{oo} \), as following equation:

\[
\mu^1_{oo} = \mu^1_{oo} \land \mu^1_{oo} \land \cdots \land \mu^1_{oo}
\]

(11)

As mentioned above, by deciding the depth and the base width of concave, PMF \( \mu^1_{oo} \), which aims to early starting of avoidance behavior and prompt the direction of the velocity vector to be far from obstacle direction in response to the fast-moving obstacle, is generated.

### 3.2.2 PMF for a goal

To head to the goal, a PMF \( \mu^1_{go} \) shaped like triangle as shown in Fig.6 (b). \( \mu^1_{go} \) is specified by \( g_o, g_b, \varphi_{r,g} \). As a measure to decide how much the robot want to head to the goal, \( g_o \) is defined as the height of the triangular PMF. As a measure to decide how much the robot is allowed to back away from obstacles, \( g_b \) is defined. \( \mu^1_{go} \) gets the maximum value as \( g_o \) at an angle of the goal direction relative to the front direction of the robot, \( \varphi_{r,g} \), and gets the minimum value as \( g_b \) at an angle of a direction opposite to the goal direction. \( g_o \) and \( g_b \) are described as following equations:
where \( \| \mathbf{r}_{o} \| \) is an absolute value of the position vector of the goal relative to the robot. \( \varepsilon \) and \( \eta \) are constants. If \( \| \mathbf{r}_{o} \| \) is below \( \varepsilon \), \( g_{a} \) is defined. The shorter the distance between the obstacle and the robot is, the smaller \( g_{a} \) becomes. The robot decelerates and stops stably.

### 3.3 Calculation of translational command velocity

The proposed method employs fuzzy inference to calculate the current command velocity vector. Specifically, The PMF \( \mu_{o}^{t} \) and the PMF \( \mu_{g}^{t} \) are integrated by fuzzy operation into a mixed PMF \( \mu_{\text{mix}}^{t} \) as shown in Fig. 7. \( \mu_{\text{mix}}^{t} \) is an algebraic product of \( \mu_{o}^{t} \) and \( \mu_{g}^{t} \) as following equation:

\[
\mu_{\text{mix}}^{t} = \mu_{g}^{t} \cdot \mu_{o}^{t}.
\]

![Fig. 7. A mixed PMF for translational motion](image)

Finally, by defuzzifier, the command velocity vector is calculated as a traveling direction \( \varphi_{\text{out}} \) and an absolute value of the reference speed of the robot base on the mixed PMF \( \mu_{\text{mix}}^{t} \). \( \varphi_{\text{out}} \) is decided as the direction which makes the PMF \( \mu_{\text{mix}}^{t} (\varphi) \) maximum. Based on \( \varphi_{\text{out}} \), \( v_{\text{out}} \) is calculated as following equation:

\[
v_{\text{out}} = \mu_{\text{mix}}^{t} (\varphi_{\text{out}}) (v_{\text{max}} - v_{\text{min}}) + v_{\text{min}}.
\]

where \( \mu_{\text{mix}}^{t} (\varphi_{\text{out}}) \) is the mixed PMF for translational movement corresponding to the \( \varphi_{\text{out}} \). \( v_{\text{max}} \) and \( v_{\text{min}} \) are configured in advance respectively as higher and lower limit of the robot speed.

### 3.4 PMF for rotational motions

#### 3.4.1 PMF for obstacles

In order to enable a wide-robot to decide the appropriate angle of direction for obstacle avoidance in real time, PMF \( \mu_{c}^{t} \) is generated based on a PMF \( \mu_{c}^{t} \), which considers the environmental information, and another PMF \( \mu_{c}^{t} \) as following equation:
\[ \mu_r^o = \mu_r^o - \mu_r^r. \]  

(16)

\( \mu^r \) is generated based on the information of distances from the center of the robot to obstacles corresponding to all directions, as shown in Fig. 8. The relative information of the distances is obtained by use of range sensors such as the laser range finder, the ultra sonic sensors or the infrared sensors. \( \mu_r^r \) is generated based on the capsule case which is introduced in 2. is calculated with Eq. (1) as following equation:

\[ \mu_r^r(\varphi) = \frac{1}{\alpha} \frac{1}{\varphi + n}. \]  

(17)

Fig. 8. PMFs for rotational motions: \( \mu_r^r \) is a PMF for an obstacle (a), \( \mu_g^r \) is a PMF for a goal (b)

The aim of the PMF \( \mu_r^r \) in (16) is to search an orientation angle of the robot which enables the distance between a point on capsule case and each obstacle to maximize, by turning front or back side of the robot on the direction that there is a closest point to each obstacle. By considering the capsule case, a design of PMF can deal with the width of the robot for rotational motion.

3.4.2 PMF for a goal

In order to turn front on the goal direction or traveling direction if there is no obstacle to avoid, PMF for a goal for rotational motion is generated as \( \mu_g^r \). This shape is decided in same way with \( \mu_g^r \), by using (12), (13).

3.5 Calculation of rotational command velocity

As for the rotational movement, like the translational movement, the proposed method employs fuzzy inference to calculate the current rotational command velocity vector. Specifically, The PMF \( \mu_g^r \), which considers the own condition by using Eq. (1), and the PMF \( \mu_r^r \), which is to head to the goal, are integrated by fuzzy operation into a mixed PMF \( \mu_{mix}^r \) as shown in Fig. 9. \( \mu_{mix}^r \) is an algebraic product of \( \mu_r^r \) and \( \mu_g^r \) as following equation:

\[ \mu_{mix}^r = \mu_g^r \cdot \mu_r^r. \]  

(18)

Finally, by defuzzifier, the command velocity vector is calculated as a traveling direction \( \varphi_{ori} \) and an absolute value of the reference speed of the robot base on the mixed PMF \( \mu_{mix}^r \). \( \varphi_{ori} \) is decided as the direction \( \varphi \), which makes a following function \( h(\varphi) \) minimum.
Fig. 9. A mixed PMF for rotational motion

where \( \zeta \) is the parameter to avoid choosing undesirable \( \phi \) caused by such as noises on the sensor data. Based on \( \phi_{ori} \), \( \omega \) is calculated as following equation:

\[
\omega = \text{sgn}(\phi_{ori})
\]

(20)

where \( \mu_{\text{mix}}^{t}(\phi_{ori}) \) is the mixed PMF for translational movement corresponding to the \( \phi_{ori} \), \( \omega_{\text{max}} \) and \( \omega_{\text{min}} \) are configured in advance respectively as higher and lower limit of the rotational speed of the robot.

3.6 Calculation of wheel speeds

To realize the movement, in this study, an omni-directional platform is employed for an autonomous mobile robot. The command velocity vector is realized by four DC motors and omni wheels using following equations:

\[
v_r^x = \|v_{out}\|\cos\phi_{out} \cdot \mu_c (\phi) \\
v_r^y = \|v_{out}\|\sin\phi_{out} \cdot \mu_c (\phi)
\]

(21)  (22)

\[
\begin{pmatrix}
\nu_1^w \\
\nu_2^w \\
\nu_3^w \\
\nu_4^w
\end{pmatrix} =
\begin{pmatrix}
\cos\delta & \sin\delta & R \\
-\cos\delta & \sin\delta & R \\
-\cos\delta & -\sin\delta & R \\
\cos\delta & -\sin\delta & R
\end{pmatrix}
\begin{pmatrix}
\nu_r^x \\
\nu_r^y \\
\nu_r^z \\
\omega
\end{pmatrix}
\]

(23)

where \( v_{out} \) and \( \omega \) are respectively current command translational velocity vector and rotational speed. \( \delta \) is an angle of gradient for each wheel. \( R \) is a half of a distance between two catawampus wheels. \( v_i^w \) is a command movement speed of each \( i \)-th wheel.

4. Simulation results

The effectiveness of the proposed method was verified by numerical simulations intended for omni-directional autonomous mobile robots. As postulates, the robot supposed to be...
Fig. 10. An omni-directional platform with wide body able to detect obstacles and has information about the relative position vector. The measuring range was assumed to be 4.0m at all directions. Each parameter was as follows: The wide-robot size was assumed as $L=0.4m$, $W=1.0m$, which are in Fig.10. Considering this $L$ and $W$, $C_a$, $C_L$ and $C_R$ in Fig.2 were all set at 0.3m. $r_o, d_r$ in Fig.5 were both set at 0.3m. Consequently $D=0.9m$ in Eq.(9). $\alpha$ in Eq. (8) was 4.0m. $\eta$ in Eq. (13) was 0.2. $\epsilon$ in Eq. (12) was 1.0m. $v_{max}$ and $v_{min}$ in Eq. (15) were respectively 0.5m/s and 0.0m/s.

4.1 Performance of capsule case

In this section, the effectiveness of using capsule case and the design method of PMF based on the capsule case are verified, by comparing the results of chosen direction of movement at following two different situations about the orientation angle for a wide-robot. As common assumption, the positions of the robot and two obstacles were immobilized on each point respectively (1.0m, 2.0m), (1.5m, 0.5m) and (3.0m, 2.0m), as shown in Fig.11.

Fig. 11. Simulation results : in Situation I (a), the robot cannot find the direction between the two obstacles, in Situation II (b), the robot can find the direction between the two obstacles.
4.1.1 Situation I
The orientation angle of the robot was fixed to \(-\pi/4\) radian clockwise from x-axis on the absolute coordinate. Therefore, the robot faced to a goal point, as shown in Fig.11(a), however, the chosen direction of the current movement of the robot was calculated as \(-1.35\) radian, which was clockwise from the front direction of the robot, as shown in Fig.11(a).

This value of the chosen direction was calculated based on the mixed PMF \(\mu_{mix}^i\) in this situation as shown in Fig.12(a). As a result, the robot chose the direction, which the robot would go the roundabout route.

(a) Situation I : the robot cannot find the direction between the two obstacles

(b) Situation II: the robot can find the direction between the two obstacles

Fig. 12. Aspects of mixed PMFs for translational motion in two different situations: (a) is in relation to the situation of Fig.11(a), and (b) is in relation to Fig.11(b).

Fig. 13. A simulation result of Method I (conventional) : The robot not using capsule case didn’t succeed in going through between two divided objects
4.1.2 Situation II

The orientation angle of the robot is fixed to $\pi/4$ radian on the absolute coordinate. As contrasted to Situation I, the robot didn’t faced to a goal point, as shown in Fig.11(b), however, the chosen direction of the current movement of the robot was calculated as 1.37 radian, which was clockwise from the front direction of the robot, as shown in Fig.11(b). This value of the chosen direction was calculated based on the mixed PMF $\mu_{\text{mix}}$ in this situation is shown in Fig.12(b). As a result, the robot chose the direction, which the robot would take a shorter route without collision.

![Fig. 14. A simulation result of Method II (proposed): The robot using capsule case succeeded in going through between two objects with translational and rotational motion in real time.](image1)

Through these two results, the effectiveness of the capsule case is confirmed. The wide-robot can decide the direction of translational motion with considering the own orientation, goal position and obstacle positions simultaneously in real time.

4.2 Capability of going through between objects

The effectiveness of the proposed method was tested by comparing two design methods, I and II, based on PMF, for obstacle avoidance problem. Start and goal point of the robot are respectively (0.0m, 0.0m) and (8.0m, 0.0m). The trajectory of the robot and the aspects considering the orientation angle on the position every 1 second are plotted in Fig. 13 and Fig. 14. Obstacles positions are respectively (2.5m, -1.8m), (2.5m, -1.2m), (2.5m, 1.2m), (2.5m, 1.8m). In Method I, as a conventional method, the robot was regarded as a circle with radius 0.6m. In Method II, as a proposed method, the capsule case was used and rotational motion of the robot was taken into consideration. When the Method I was used, the robot was based on the maximum radius and did not take into consideration the rotational motion. Therefore, the robot did not succeed in going between two objects as shown in Fig. 13. While the robot did not collide with the obstacles, the robot did not get to the goal.

On the other hand, in the Method II, the capsule case and real-time control based on FPM were used. As shown in Fig. 14, the robot performed translational and rotational motions simultaneously and succeeded in going between two objects. In addition, the robot succeeded in getting to the goal with the orientation angle 0 radian using PMF for rotational motion.
The effectiveness of the proposed method was verified also by simplified experiments using omni-directional autonomous mobile robots as shown in Fig. 15. In each picture of the Fig. 15, aspects of the robot every 1 second are plotted. The robot recognized environment by the omni-directional camera. A position of a goal and that of obstacles relative to the robot were calculated by extracting features in images based on objects’ colours. A ball was assumed as the goal and columns were assumed as obstacles, as shown in Fig. 15. A dashed circle enveloping a column in Fig. 15 corresponds to a dashed circle in Fig. 5. As shown in Fig. 15(a), the robot was not able to between two objects without the capsule case. However, as shown in Fig. 15(b), the robot with the capsule case performed translational and rotational motion simultaneously in real time and succeeded in going between two obstacles.

These results showed that motion control without a capsule case made it difficult for the robot to go between two objects due to the largest radius of the robot, even if it would be able to go through by using its shortest radius. Applying the capsule case to a wide robot enhances the possibility of going between two objects.

5. Conclusion
In this paper, the real time control method of simultaneously translational and rotational motions for an autonomous mobile robot, which is horizontally long, has been introduced. This method employs omni-directional platform for the drive system and is based on the fuzzy potential method (FPM). The novel design method of potential membership function (PMF), which is considered the width of the robot by using the capsule case, has been shown. According to this proposed method, the wide-robot can decide the current direction of translational motion to avoid obstacles safely by using capsule case. In addition, by controlling the rotational motion in parallel with the translational motion in real time, the wide-robot can go through narrow distance between two objects. The effectiveness has been verified by numerical simulations and simplified experiments. It has been shown that the proposed method enables simultaneous control of the translational and rotational velocity within the framework of FPM.
5. References


www.intechopen.com
Numerical Analysis – Theory and Application is an edited book divided into two parts: Part I devoted to Theory, and Part II dealing with Application. The presented book is focused on introducing theoretical approaches of numerical analysis as well as applications of various numerical methods to either study or solving numerous theoretical and engineering problems. Since a large number of pure theoretical research is proposed as well as a large amount of applications oriented numerical simulation results are given, the book can be useful for both theoretical and applied research aimed on numerical simulations. In addition, in many cases the presented approaches can be applied directly either by theoreticians or engineers.

How to reference
In order to correctly reference this scholarly work, feel free to copy and paste the following:

© 2011 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License, which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.