William M. Spears
Diana F. Spears *Editors*

Physico-mimetics

Physics-Based Swarm Intelligence

Springer

Physicomimetics

William M. Spears • Diana F. Spears
Editors

Physicomimetics

Physics-Based Swarm Intelligence

Springer

Editors
Dr. William M. Spears
Swarmotics LLC
Laramie, WY
USA
wspears@swarmotics.com

Dr. Diana F. Spears
Swarmotics LLC
Laramie, WY
USA
dspears@swarmotics.com

ISBN 978-3-642-22803-2
e-ISBN 978-3-642-22804-9
DOI 10.1007/978-3-642-22804-9
Springer Heidelberg Dordrecht London New York

Library of Congress Control Number: 2011942477

ACM Classification (1998): I.2, I.6, J.2

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

To Daisy

Preface

We are used to thinking of the world in a centralized and hierarchical manner. Governments and businesses rely on organizations with someone "at the top" who collects information and issues orders that trickle down the hierarchy until they reach the rest of us. Even portions of our economic system work in this fashion. The reason organizations exist is because they work well in many situations. But there is another view of the world that is entirely different. This view starts at the "bottom," and realizes that much of the organization that we see does not stem from centralized control, but emerges from the local interactions of a multitude of entities (as with insects, people, vehicles, and the movement of money). These multitudes are swarms. Standard approaches to understanding swarms rely on inspiration from biology. These are called "biomimetic" approaches. In this book, we focus on a different inspiration, namely, physics. We refer to physics-based swarm approaches as "physicomimetics." Both approaches are complementary, but physics-based approaches offer two unique advantages. The first is that these approaches capture the notion that "nature is lazy." This means that physical systems always perform the minimal amount of work necessary. This is very important for swarm robotics, because robots are always limited by the amount of power they have at their disposal. The second advantage is that physics is the most predictive science, and it can reduce complex systems to amazingly simple concepts and equations. These concepts and equations codify emergent behavior and can be used to help us design and understand swarms.

This book represents the culmination of over 12 years of work by numerous people in the field of swarm intelligence and swarm robotics. We include supplemental material, such as simulation code, simulation videos, and videos of real robots. The goal of this book is to provide an extensive overview of our work with physics-based swarms. But we will not do this in the standard fashion. Most books are geared toward a certain level of education (e.g., laymen, undergraduates, or researchers). This book is designed to "grow with the reader." We start with introductory chapters that use simple but powerful graphical simulations to teach elementary concepts in physics and swarms. These are suitable for junior and senior high school students. Knowledge of algebra and high school physics is all you need to understand this material. However, if you are weak in physics, we provide a chapter, complete with simulations, to bring you back up to speed. In fact, even if you have had physics already, we recommend that you read this chapter—because the simulations provide insights into physics that are difficult to achieve using only equations and standard high school physics textbooks.

All you need is a computer to run the simulations. They can be run directly on your machine or through your web browser. You do not need to have had any programming courses. But if you have had a programming course, you will be ready to modify the simulations that come with this book. We provide an introductory chapter to explain the simple simulation language that we use throughout. Suggestions for modifications are included in the documentation with the simulations.

The middle of the book is most appropriate for undergraduates who are interested in majoring in computer science, electrical computer engineering, or physics. Because we still use simulations, these chapters generally require only algebra and an understanding of vectors. A couple of chapters also require basic calculus (especially the concepts of derivatives and integrals) and elementary probability. If you don't know anything about electrical computer engineering, that is fine. You can merely skim over the hardware details of how we built our robots and watch the videos that also come with this book. But if you have had a course or two, much of this material will be quite accessible.

The final sections contain more advanced topics suitable for graduate students looking for advanced degree topics, and for researchers. These sections focus on how to design swarms and predict performance, how swarms can adapt to changing environments, and how physicomimetics can be used as a function optimizer. However, even here most of the chapters require little mathematics (e.g., only two require knowledge of linear algebra and calculus).

It is important to point out that this is a new and rapidly developing field. Despite the fact that the authors share a surprisingly consistent vision, we do not always view swarms in precisely the same way. This should not be a cause for concern. Unlike Newtonian physics or mathematics, which have been developed for hundreds of years, physicomimetics is relatively new and does not yet have entirely consistent terminology and notation. But all this means is that not everything is cast in stone—we are at the beginning of an adventure. You are not learning old knowledge. Instead, you are seeing how science progresses in the here and now. We hope you enjoy the journey.

Laramie, Wyoming *William M. Spears*
September 2011 *Diana F. Spears*

Acknowledgements

This book would not have been possible without the hard work, creativity, and perseverance of our numerous students. Quite a few of them are contributors to this book, and we thank them all. Suranga Hettiarachchi, Paul M. Maxim, and Dimitri Zarzhitsky earned Ph.D.'s working on the projects described in this book. In addition, Dimitri Zarzhitsky provided valuable proofreading of portions of this book.

Thanks to Adam Sciambi for our first graphical user interface for artificial physics (physicomimetics). Adam was a high school student who participated in the Science and Engineering Apprentice Program at the Naval Research Laboratory. During July 2000, he converted our original C code into Java, including two-dimensional and three-dimensional simulations. Brian Zuelke and Vaibhav Mutha subsequently extended Adam's simulations.

We appreciate the collaboration with Oleg Sokolsky and Insup Lee of the University of Pennsylvania on an early project that consisted of combining artificial physics with monitoring and checking.

We are grateful to David Walrath and Ben Palmer of the University of Wyoming for the creation of a spheroidal radio-controlled vehicle that we tested for outdoor use. Ben did this for his senior design project. Thanks also to Ella Wellman, who helped us with our early simulation experiments on chain formations.

We thank Alan Schultz of the Naval Research Laboratory for the use of his Lego robotics kits from the KISS Institute of Practical Robotics. Thanks to Paul Hansen for using these kits to provide us with the first robots that we used for our initial experiments. These early robots were invaluable for testing artificial physics, and they served as platforms for tests of our trilateration hardware and for preliminary chemical plume tracing experiments. John Schabron of the Western Research Institute provided valuable guidance on the calibration and testing of our chemical sensors.

Furthermore, we would like to thank the students who took our course on artificial intelligence and robotics: Nathan Horter, Christoph Jechlitschek, Nathan Jensen, Nadya Kuzmina, Glenn McLellan, Vaibhav Mutha, Markus

Petters, Brandon Reed, and Peter Weissbrod. They worked hard, pushing the Lego robots to their limits, and provided a lot of data for us on the proper use of Sharp infrared sensors and shaft encoders. One student thought it was "by far one of the most entertaining projects I've ever worked on." Another said, "I'll probably remember more about this class than most other classes I've taken, in part because I got to work in a group and discuss how to proceed in solving problems in both software and hardware designs which I really don't get a chance to do in other classes, and, it was just a really fun project to do." Students found that working with actual hardware provided them with a very real appreciation for how difficult it is to sense an environment accurately and to respond appropriately. Given that modern operating systems require almost one gigabyte of memory, it was exciting to see how much could be accomplished with only 32 kilobytes! Student enthusiasm and energy always provided us with a deep sense of satisfaction.

We appreciate the guidance from Jerry Hamann of the University of Wyoming on designing the trilateration hardware, which was key to all of our later experiments. Numerous students were instrumental in building and improving the hardware, especially Rodney Heil, Thomas Kunkel, and Caleb Speiser. These three students earned master's degrees while working on these projects. Rodney Heil also solved the initial trilateration equations, thus providing an extremely elegant solution.

Special thanks to William Agassounon and Robert Welsh from Textron Defense Systems for funding and support for our chemical plume tracing research and our next-generation trilateration system.

We wish to thank James Partan, who was a visiting Ph.D. student from the University of Massachusetts, Amherst. Despite the fact that robotics was not Jim's area of expertise, he shared an interest with respect to disruption-tolerant wireless networks. He always provided excellent insights every time we used him as a sounding board.

Most of the NetLogo simulations provided with this book were originally written in StarLogo, which is a similar language. We thank Eric Klopfer of the Massachusetts Institute of Technology for his advice and feedback on StarLogo.

Finally, we extend our appreciation to our supportive European colleagues, especially Erol Şahin, Alan Winfield, and Alcherio Martinoli. Those of you who are interested in learning still more about swarms will find that the proceedings of the First and Second International Workshops on Swarm Robotics will provide you with a thorough introduction to the field [202, 203].

Contents

Part I Introduction

Part II Robotic Swarm Applications

Part III Physicomimetics on Hardware Robots

Part IV Prediction, Adaptation, and Swarm Engineering

Part V Function Optimization

List of Contributors

Richard Anderson-Sprecher
Department of Statistics, University of Wyoming, Laramie, Wyoming, USA, e-mail: sprecher@uwyo.edu

Thomas B. Apker
NRC/NRL Postdoctoral Fellow, Washington DC, USA, e-mail: apker@aic.nrl.navy.mil

Andrea Bravi
Dynamical Analysis Laboratory, University of Ottawa, Ottawa, Ontario, Canada, e-mail: a.bravi@uottawa.ca

Paolo Corradi
Human Space Flight and Operations, European Space Agency, Noordwijk, The Netherlands, e-mail: paolo.corradi@sssup.it

Chris Ellis
Department of Electrical Engineering and Computer Science, University of Central Florida, Orlando, Florida, USA, e-mail: chris@cs.ucf.edu

Charles Lee Frey
Harbor Branch Oceanographic Institute, Ft. Pierce, Florida, USA, e-mail: cfrey@hboi.fau.edu

Derek T. Green
Department of Computer Science, University of Arizona, Tucson, Arizona, USA, e-mail: dtgreen@arizona.edu

Suranga Hettiarachchi
Department of Computing Science, Indiana University Southeast, New Albany, Indiana, USA, e-mail: suhettia@ius.edu

Christer Karlsson
Department of Mathematical and Computer Sciences, Colorado School of Mines, Golden, Colorado, USA, e-mail: ckarlsso@mines.edu

Sanza Kazadi
Jisan Research Institute, Alhambra, California, USA, e-mail: skazadi@jisan.org

Wesley Kerr
Department of Computer Science, University of Arizona, Tucson, Arizona, USA, e-mail: wkerr@email.arizona.edu

Aleksey Kletsov
Department of Physics, East Carolina University, Greenville, North Carolina, USA, e-mail: kletsov@gmail.com

Paul Maxim
Wyoming Department of Transportation, Cheyenne, Wyoming, USA, e-mail: paul.maxim@dot.state.wy.us

Arianna Menciassi
CRIM Laboratory, Scuola Superiore Sant'Anna, Pisa, Italy, e-mail: arianna.menciassi@sssup.it

Mitchell A. Potter
US Naval Research Laboratory, Washington DC, USA, e-mail: mpotter@aic.nrl.navy.mil

Antons Rebguns
Department of Computer Science, University of Arizona, Arizona, USA, e-mail: anton@email.arizona.edu

Florian Schlachter
Applied Computer Science – Image Understanding, University of Stuttgart, Stuttgart, Germany, e-mail: Florian.Schlachter@ipvs.uni-stuttgart.de

Diana F. Spears
Swarmotics LLC, Laramie, Wyoming, USA, e-mail: dspears@swarmotics.com

William M. Spears
Swarmotics LLC, Laramie, Wyoming, USA, e-mail: wspears@swarmotics.com

Ying Tan
Taiyuan University of Science and Technology, Taiyuan, Shanxi, China, e-mail: tanying1965@gmail.com

David Thayer
Physics and Astronomy Department, University of Wyoming, Laramie, Wyoming, USA, e-mail: drthayer@uwyo.edu

R. Paul Wiegand
Institute for Simulation and Training, University of Central Florida, Orlando, Florida, USA, e-mail: wiegand@ist.ucf.edu

Edith A. Widder
Ocean Research and Conservation Association, Ft. Pierce, Florida, USA,
e-mail: ewidder@oceanrecon.org

Liping Xie
Taiyuan University of Science and Technology, Taiyuan, Shanxi, China,
e-mail: xieliping1978@gmail.com

Dimitri V. Zarzhitsky
Jet Propulsion Laboratory, California Institute of Technology, Pasadena,
California, USA, e-mail: Dimitri.Zarzhitsky@jpl.nasa.gov

Jianchao Zeng
Taiyuan University of Science and Technology, Taiyuan, Shanxi, China,
e-mail: zengjianchao@263.net

Acronyms

AP	Artificial physics
APO	Artificial physics optimization
ASV	Autonomous surface vehicle
AUV	Autonomous underwater vehicle
BP	Bathyphotometer
BPT	Biological plume tracing
CFD	Computational fluid dynamics
CFL	Chain formation list
CPT	Chemical plume tracing
DAEDALUS	Distributed agent evolution with dynamic adaptation to local unexpected scenarios
DG	Density gradient
DMF	Divergence of chemical mass flux
EA	Evolutionary algorithm
EL	Evolutionary learning
IP	Internet protocol
EEPROM	Electrically erasable programmable read-only memory
FOFE	First-order forward Euler integration
GDMF	Gradient of the divergence of the mass flux
GMF	Gradient of the mass flux
GPS	Global Positioning System
GUI	Graphical user interface
HABS	Harmful algal blooms
HBOI	Harbor Branch Oceanographic Institute
HLS	Hyperbolic localization system
I2C	Inter-Integrated Circuit
IR	Infrared
KL	Kullback–Leibler
KT	Kinetic theory
LCD	Liquid crystal display
LOPF	Local oriented potential fields

LOS	Line-of-sight
MAV	Micro-air vehicle
MAXELBOT	Robot built by Paul M. Maxim and Tom Kunkel
MDS	Multi-drone simulator
MST	Minimum spanning tree
MSE	Mean squared error
NAVO	Naval Oceanographic Office
NRL	Naval Research Laboratory
NSF	National Science Foundation
ONR	Office of Naval Research
PCB	Printed circuit board
PIC	Programmable intelligent computer
PSO	Particle swarm optimization
PSP	Paralytic shellfish poisoning
PVC	Polyvinyl chloride
RF	Radio frequency
RL	Reinforcement learning
SRF	Sonic range finder
TCP/IP	Transmission control protocol / Internet protocol
UAV	Unmanned aerial vehicle
VOC	Volatile organic compound
VSP	Voith–Schneider propeller
XSRF	Experimental sonic range finder

NetLogo Syntax Glossary

This glossary provides a useful (albeit incomplete) glossary of NetLogo syntax.

`ask patch x y [a]`	Ask patch at (x, y) to execute a.
`ask patches [a]`	Ask all patches to execute a.
`ask turtle n [a]`	Ask turtle n to execute a.
`ask turtles [a]`	Ask all turtles to execute a.
`ca`	Kills all turtles and resets all variables to zero.
`clear-all`	Kills all turtles and resets all variables to zero.
`crt n`	Create n turtles, numbered 0 to $n-1$.
`end`	The keyword that ends the definition of NetLogo procedures.
`home`	Send all turtles to $(0, 0)$.
`if (a) [b]`	If condition a is true execute b.
`ifelse (a) [b] [c]`	If condition a is true execute b, else execute c.
`pd`	Set the pen to be down.
`pen-down`	Set the pen to be down.
`pen-up`	Set the pen to be up.
`pu`	Set the pen to be up.
`random n`	Returns a random integer uniformly from 0 to $n - 1$.
`set a b`	Set the value of variable a to b.
`set color white`	Set the color of a turtle to white.
`set heading a`	Set the heading of a turtle to a.
`to`	The keyword that starts the definition of NetLogo procedures.
`[xcor] of turtle 1`	The x-coordinate of turtle number one.

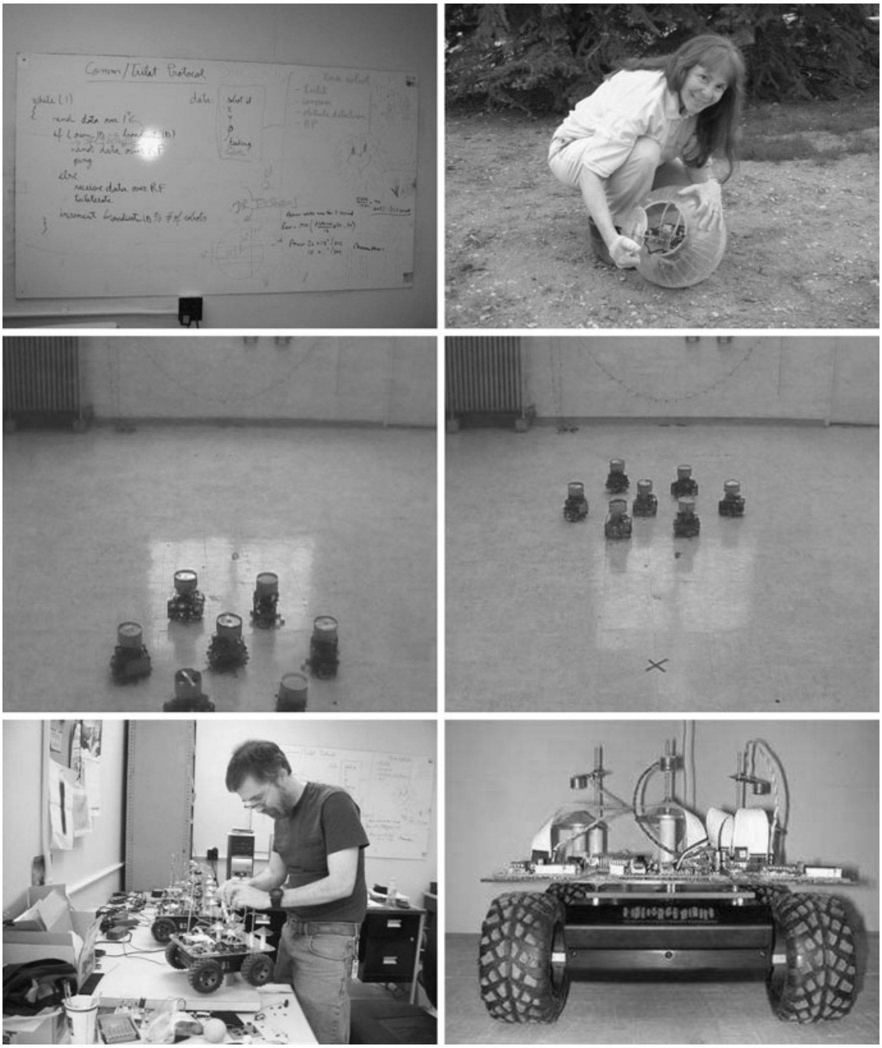
NetLogo Parameters Glossary

This glossary provides a useful (albeit incomplete) glossary of parameters used in our NetLogo simulations.

angular_mom	The angular momentum of the system.
breed	The breed of an agent.
center_of_mass_x	The x-component of the center of mass of the system.
center_of_mass_y	The y-component of the center of mass of the system.
color	The color of an agent.
dv	The change in velocity of an agent.
dvx	The x-component of the change in velocity of an agent.
dvy	The y-component of the change in velocity of an agent.
DeltaT	The time granularity of the simulation.
deltax	The x-component of distance between two agents.
deltay	The y-component of distance between two agents.
F	The force on an agent.
FMAX	The maximum pair-wise force magnitude.
FR	The amount of friction in the system.
Fx	The x-component of the force on an agent.
Fy	The y-component of the force on an agent.
G	The gravitational constant in the system.

hood	The neighbors (neighborhood) of an agent.
heading	The heading of an agent.
k	The spring constant.
ke	The kinetic energy of an agent.
lm	The linear momentum of an agent.
lmx	The x-component of the linear momentum of an agent.
lmy	The y-component of the linear momentum of an agent.
mass	The mass of an agent.
omega	The initial amount of rotation in the system.
pcolor	The color of a patch.
pe	The potential energy of an agent.
r	The r distance between two agents.
total_energy	The total energy of the system.
total_ke	The total kinetic energy of the system.
total_lm	The total linear momentum of the system.
total_lmx	The x-component of the total linear momentum of the system.
total_lmy	The y-component of the total linear momentum of the system.
total_pe	The total potential energy of the system.
v	The velocity of an agent.
VMAX	The maximum velocity of an agent.
vx	The x-component of the velocity of an agent.
vy	The y-component of the velocity of an agent.
who	Returns the id of the turtle.
xcor	The x-coordinate of an agent or patch.
ycor	The y-coordinate of an agent or patch.

Part I
Introduction



Chapter 1
Nature Is Lazy

William M. Spears

"...sudden I wav'd My glitter falchion, from the sanguine pool Driving th' unbody'd host that round me swarm'd" (1810) [29]

"...ſince then it is called a ſwarm of bees, not ſo much from the murmuring noiſe they make while flying, as the manner in which they connect, and join themſelves together at that remarkable time of ſwarming" (1783) [135]

1.1 What Are Swarms?

No one knows the exact origin of the word "swarm." Most sources refer to the Old English word "swearm." Samuel Johnson's 1804 dictionary defined a swarm as "a great number of bees; a crowd" [108]. Similarly, in 1845 a swarm was defined as a large number of persons or animals in motion [178]. This agrees with the use of the word in the two quotations above. The first is from a translation of Homer's Odyssey [29]. The second is from a book on English etymology [135].

The most frequent connotation of "swarm" is with respect to bees, animals and people. And, not surprisingly, much of the swarm research is inspired by biology. This research is referred to as "biomimetics" or "biomimicry." These words are from the Greek *bios* ($\beta\acute{\iota}o\varsigma$), meaning "life," and *mimesis* ($\mu\acute{\iota}\mu\eta\sigma\iota\varsigma$), which means "imitation." Hence much of swarm research focuses on the imitation of live organisms such as birds, fish, insects, and bacteria.

This book examines a different approach to swarms, namely a "physicomimetics" approach. This word is derived from *physis* ($\varphi\acute{\upsilon}\sigma\iota\varsigma$), which is Greek for "nature" or "the science of physics." Physics involves "the study of matter and its motion through spacetime, as well as all related concepts, including energy and force" [265]. Hence we will focus on swarms of matter particles that are subject to forces. Depending on the application the particles can act as robots in an environment, agents solving a task, or even as points in a high dimensional space. We will show how various forms of energy play a role in how the particles move.

William M. Spears
Swarmotics LLC, Laramie, Wyoming, USA, e-mail: wspears@swarmotics.com

W.M. Spears, D.F. Spears (eds.), *Physicomimetics*,
DOI 10.1007/978-3-642-22804-9_1,