Proceedings of ECCS 2014
European Conference on Complex Systems
Springer Proceedings in Complexity

Tangible Networks: A toolkit for exploring network science

Espen Knoop[1,2], Edmund Barter[1], Alonso Espinosa Mireles de Villafranca[1], Antoni Matyjaszkiewicz[1], Christopher McWilliams[1], and Lewis Roberts[1]

[1] Department of Engineering Mathematics, University of Bristol, University Walk, Bristol BS8 1TR, UK
[2] Bristol Robotics Laboratory, T Block, Frenchay Campus, Coldharbour Lane, Bristol, BS16 1QY, UK
espen.knoop@bristol.ac.uk

Abstract. We present Tangible Networks (TN), a novel electronic toolkit for communicating and explaining concepts and models in complexity sciences to a variety of audiences. TN is an interactive hands-on platform for visualising the real-time behaviour of mathematical and computational models on complex networks. Compared to models running on a computer, the physical interface encourages playful exploration. We discuss the design of the toolkit, the implementation of different mathematical models and how TN has been received to date.

Keywords: Interactive model, visualisation, science communication, tangible interface, complex systems, mathematical modelling.

1 Introduction

Our work focuses on communicating ideas from complexity science to a non-specialist audience. Academics are frequently required to communicate their research to funding bodies. Since the Wolfendale committee [26], there has been an increasing drive to communicate research to the public and promote dialogue [5, 27]. We believe that public engagement is of particular importance to complex systems research because of its relevance to a wide range of systems in nature, engineering and social sciences. Drawing on our experience of public engagement [3], we have developed Tangible Networks (TN) for communicating key concepts from complexity science, and more generally facilitating learning.

Complexity scientists make extensive use of tools which can be unfamiliar to non-specialists, including advanced mathematical techniques and computer algorithms. From our experience it can be a challenge to effectively communicate concepts in complex systems without referring to these tools. Furthermore, mathematical and algorithmic models are becoming increasingly pervasive in today's society. Therefore, making these models more accessible and engaging is an important task.

One effective way of communicating such concepts is by using interactive models, where users can perturb the simulation and observe how the dynamics are affected. For example, *NetLogo*[3] is an interactive visualisation software package developed for exploring agent-based systems in real time. *Danceroom Spectroscopy* [12, 19] is an interactive model of molecular dynamics where the visualisation of the moving molecules is projected on a screen and the bodies of users become "energy landscapes" that directly affect the forces on the molecules. Danceroom Spectroscopy has been used as an art installation, in dance performances, for education and also for research. As well as teaching, these platforms are useful for raising awareness of mathematical modelling.

Mathematical models are generally presented on a computer. The idea of Tangible Interfaces [15] is to interact with the digital world by manipulating physical objects. It has been argued that Tangible Interfaces encourage playful learning and creative exploration, are more accessible and are well suited for collaboration [18]. There are a number of educational toolkits for teaching electronics (e.g. *LittleBits* [4]) and robotics (e.g. *cubelets*, previously *roBlocks* [24]). A tangible interface for teaching mathematics is *Smart Blocks* [11] where shapes can be constructed by snapping blocks together, and the volume and surface area of the shape is computed. Horn [14] presents a tangible tool-kit for teaching programming through interaction with physical blocks.

Taking inspiration from interactive models and tangible interfaces, we have created a tangible interactive network model. Tangible Networks makes the exploration of science and complex systems more approachable and inviting to a wider range of audiences. TN is a physical platform for network simulations that makes the key components (nodes and links, [21]) physical building blocks that can be manipulated while simulations are running, as illustrated in Fig. 1. The platform has been designed to make the network topology clearly visible and reconfigurable, so that users can get an understanding of how network topology affects behaviour.

There are several examples of robotic swarms being used to demonstrate complex network behaviour (e.g. [23]), where typically robots will communicate wirelessly with other neighbouring robots and the swarm exhibits a global behaviour from the local interactions. Although these platforms are excellent for demonstrating concepts such as swarming and emergence, it is more difficult to build a fundamental understanding of ideas such as network topology and how topology affects behaviour. We believe TN is well suited for teaching such concepts.

2 The toolkit

Many complex systems are modelled as interactions between simple agents connected in a network where collective behaviour emerges from localised interactions. With TN, we have created a physical network, where nodes and links are represented by electronic units and connecting wires respectively. Each node runs a mathematical model of the local dynamics. Users can interact with the

[3] Available online at https://ccl.northwestern.edu/netlogo/

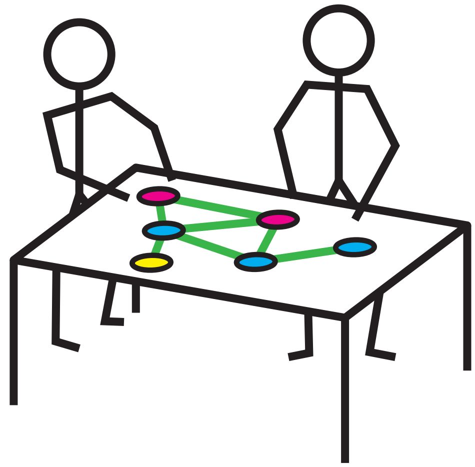

Fig. 1. We want to create a hands-on interactive visualisation of complex network models.

network and observe how the behaviour changes. The TN toolkit is shown in Fig. 2, with key features labelled.

A central objective in the development of TN was to create a simple and robust platform allowing for playful interaction. We wanted users to be able to change the network topology and add or remove nodes from the network with the simulation running.

In many real-world systems, agents only have local information. In keeping with this, we have implemented a distributed simulation where each node runs a model of the local dynamics. This implementation also improves robustness. Two nodes can only exchange information if there is a link between them. This makes the behaviour of the model more transparent — there is a very close link between the physical and mathematical structures. There are some disadvantages of distributed simulations, e.g. computing a global state of the system is difficult. However, for many systems a distributed model is highly appropriate.

In our implementation, links are directed (Fig. 2: d). This is a more general case, as undirected links can be made by combining two directed links (Fig. 2: e). The maximum degree of each node is determined by the number of physical connectors it has. In TN, nodes have a maximal in- and out-degree of three. This is the simplest case where non-trivial undirected networks can be built — with a degree of two only lines and rings would be possible.

Users can interact with the running network simulation. The topology of the network can be changed by reconfiguring the wires, and nodes can be added or removed from the network. Each node can be perturbed with a pushbutton switch (Fig. 2: a), and local parameters can be changed with a potentiometer (Fig. 2: c). The pushbutton and potentiometer can be programmed to have any function.

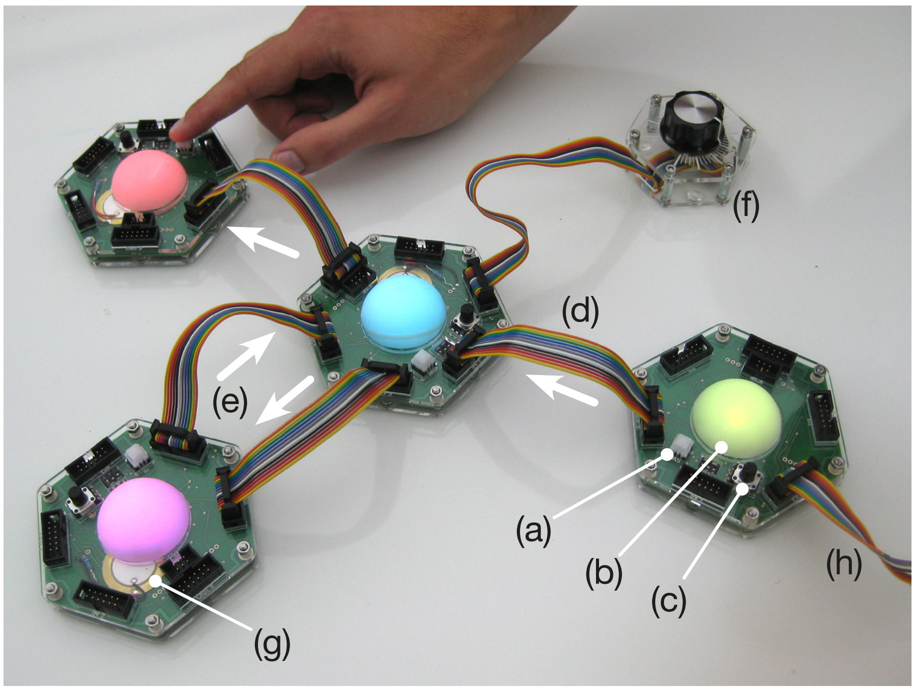

Fig. 2. The Tangible Networks toolkit. Labelled are (a) pushbutton switch; (b) glowing dome; (c) potentiometer; (d) directed link; (e) undirected link; (f) master controller; (g) piezo speaker; (h) power supply. Online version in colour.

As well as local control, users can control global properties by connecting a master controller to any one node. A master dial (Fig. 2: f) can be used to adjust a global parameter, such as the coupling parameter in models of coupled dynamical systems. This sends a continuous value to all of the nodes in the network that can be read by each node. Alternatively, a master pushbutton switch can be used for digital input such as to reset the simulation.

We require nodes to output their current state. The output must be in a form such that the collective behaviour of a large network can be easily observed. For this reason, nodes produce visual output by means of a glowing dome (Fig. 2: b). The state of each node can be visualised with the brightness and colour of the dome or by changing the frequency of brightness oscillations. Nodes can also produce sound by means of a piezo speaker (Fig. 2: g).

2.1 Technical description

Processor. An Atmel ATMega 328p microcontroller, as used in the widespread open-source Arduino platform [1], runs the local model on each node. The Arduino programming language is essentially C++, with low-level hardware control hidden in wrapper functions but still being available if required. Arduino is designed to be easy to use and is very well supported on-line, requiring no experience in microcontroller programming. We have written a library for interfacing with the TN hardware. Programs are uploaded to the TN nodes with an In-System Programming (ISP) hardware programmer.

Electronics. The state of each node is shown with a glowing dome, lit up using a high-brightness Red-Green-Blue (RGB) Light Emitting Diode (LED) behind a diffuser. LEDs are controlled with Pulse Width Modulation (PWM). Power is distributed through the network, and the power supply can be connected to any node (Fig. 2: h). Neighbouring nodes communicate via analogue signals, so each connection transmits a single real number in a limited range. This is very robust, and forces models to be simpler and more intuitive. The master controller sends an analogue voltage to all the nodes in the network. The nodes have a tactile momentary pushbutton switch; a potentiometer connected to an analogue input pin and three DIP configuration switches. Insulation Displacement Connectors (IDCs) and ribbon cables are used for the links.

Mechanical. The nodes are designed to be simple to fabricate with limited facilities. Each unit has a single Printed Circuit Board (PCB); a diffusor fabricated from a bisected table tennis ball and an enclosure comprised of two pieces of laser cut acrylic. A CAD drawing of a TN node is shown in Fig. 3.

The nodes are hexagonal, with one connector along each edge of the hexagon and a centred glowing dome. The hexagonal shape gives the units a visual identity and the symmetry lends itself well to different network topologies. Input and output connectors are alternated, so a bidirectional connection is achieved by having links to two adjacent connectors. Each node is 90 mm across, which

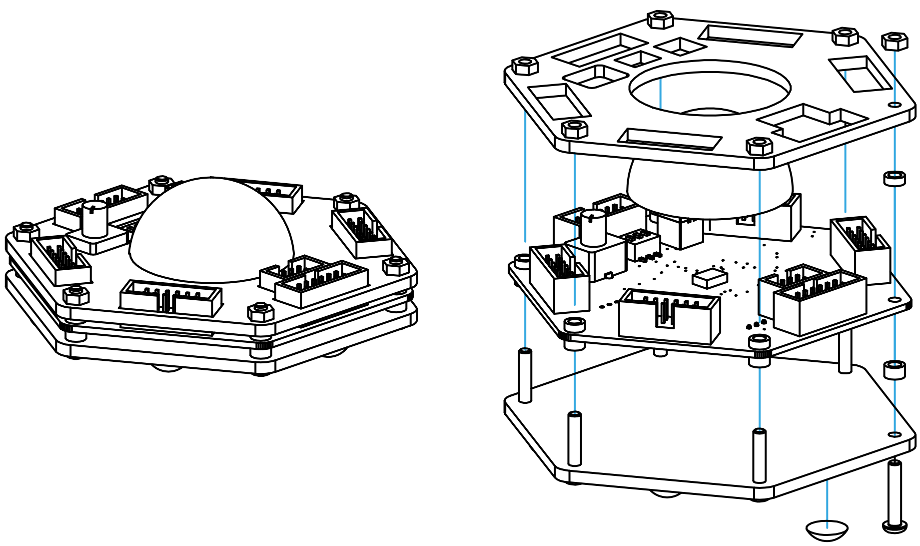

Fig. 3. CAD drawing of a TN node, in assembled and exploded states.

is sufficiently small so that it can be picked up by a child and yet large enough that the unit can be supported with the other hand when plugging in cables.

Additional functionality. For future expansion, each node also features an auxiliary input and auxiliary output as well as a serial connection. The serial connection could be interfaced to other hardware, or to a computer for plotting real-time graphs of the network dynamics. A further use of the serial port would be to combine Tangible Networks with a computer-simulated network so that the physical nodes could be connected to a large virtual network running on the computer. This would allow for much larger network simulations, where a small part of the network can be interacted with. The aux out port is connected to a general purpose pin on the microcontroller that can generate PWM signals. We have used the aux out to drive a RC servo for mechanical output, and also a piezo speaker. The auxiliary in port is connected to a general purpose pin on the microcontroller that can function as an analogue input. We have used the auxiliary input to connect a light sensor and also a single-axis accelerometer, but these are only examples of what is possible. There are a large number of examples and code snippets online for interfacing the Arduino with a range of sensors and other hardware[4].

[4] See for example https://forum.arduino.cc

3 Implemented models

Here we present some of the network models we have implemented on Tangible Networks, based on our research in complexity science. These demonstrate some key concepts from ongoing research in our group in an approachable way. We have also developed a network based game, which demonstrates the potential for TN as a tool for teaching pure mathematics through problem solving. The following model descriptions are brief, but are intended as examples of what is possible with the TN toolkit. They include discrete-time models and dynamical systems, different types of local and global interaction, and possible ways of presenting the model output.

3.1 Excitable neurons

The Fitzhugh-Nagumo (FHN) model [10, 20] is widely used to describe the spiking patterns of excitable cells such as neurons or muscle cells. Excitable cells generate a spiking electric current (an action potential) when stimulated. In our implementation, each TN node is an excitable cell. The action potential is visualised with the colour and brightness of the dome, and nodes emit sound when they spike. The pushbutton gives the cell an instantaneous stimulus, and the potentiometer sets a level of continuous stimulation. Cells also receive stimulus from their neighbours, and the global coupling parameter is set with the master dial. This introduces the idea that the specific environment that cells are in, such as the presence or absence of different substances can affect the overall dynamics. The pattern of spiking is dependent on the topology, and on the type and strength of coupling. Excitatory (positive) coupling can lead to travelling waves or synchronous oscillations, with increased coupling increasing the wave speed of propagating spiking patterns. Inhibitory (negative) coupling leads to a range of asynchronous oscillations due to post-inhibitory rebound spiking, including sustained oscillations with neighbouring nodes in antiphase for some topologies.

3.2 Synchronising oscillators

The Kuramoto model [17] describes a wide variety of synchronisation phenomena [2]. Examples include flashing fireflies [6], power grid systems [8, 9] or a conductor keeping an orchestra in time. Each TN node is an independent first order oscillator with a natural frequency that is adjusted with the potentiometer. Neighbouring nodes are coupled, and the global coupling parameter is controlled with the master dial. Stronger positive coupling increases the level of synchronisation, while negative coupling causes neighbouring nodes to oscillate in antiphase. The phase of each oscillator is shown with variations in brightness. The colour represents the local degree of synchronisation, computed as the mean phase difference with its neighbours. Oscillators can be stopped and held at a constant phase by holding down the pushbutton. Users can explore how the level of synchronisation is affected by the topology and natural frequencies.

3.3 Opinion dynamics

The majority-vote model produces qualitative results similar to the patterns of opinions in networks [7]. Each node is a person who holds one of two opinions and is influenced by their neighbours. Stubborn nodes change their opinion if more than half of their neighbours disagree with them. Fickle nodes change their opinion if at least half of the neighbours disagree with them. Nodes update their opinion in discrete timesteps. Users can set the initial opinions and stubbornness of each node, with the potentiometer. Opinions are shown as green and blue, with fickle nodes being more pale. Pressing the pushbutton toggles the opinion of that node, and the user can then observe whether this causes any further nodes to change their opinion. The simulation is reset with the master switch. The network simulation can demonstrate consensus and clustering of opinions, along with ideas such as group influence. It can also show how some nodes are more influential than others, and that the most influential nodes are not necessarily the most central or most connected nodes.

3.4 Predator-prey dynamics

The Lotka-Volterra equations describe predator-prey dynamics in ecological systems of two or more species. In our implementation, each node is a species. The brightness indicates the current population, and the colour indicates the trophic level (red: top predator, yellow: intermediate predator, green: primary producer). The trophic level is set with the DIP switches. The potentiometer sets the intrinsic growth rate of the species. Pressing the pushbutton increases the population of that species. We can demonstrate simple interactions between a single predator and a single prey that lead to oscillating populations, as well as more complex food webs. The model can be used to demonstrate meaningful ecological concepts such as competitive exclusion, apparent competition and biological pest control.

3.5 Hamiltonian paths

A Hamiltonian path is a route through the network that visits each node exactly once. In our implementation, we introduce this concept through a problem-solving exercise: Pressing the master switch resets the game, making all nodes cyan. Pressing the pushbutton on one node selects the starting node and starts the game. The current node is green, visited nodes are yellow and unvisited nodes are blue. Pressing the pushbutton on an unvisited node adjacent to the current node moves the player to that node. The game ends when there are no more possible moves, at which point visited nodes turn green and unvisited nodes turn red.

The Hamiltonian path problem is simple to solve heuristically on a small network, however the exercise can be scaffolded to ask for deeper understanding and problem solving. The game has been developed in tandem with a structured worksheet, and has been given a storyline to motivate younger children to solve the problem.

4 Discussion

We have presented Tangible Networks, a toolkit for interacting with mathematical models and communicating ideas from complex systems. A TN user can explore and learn about these systems and models even if they are unfamiliar with the underlying mathematics. TN presents the models in a more engaging and approachable way by allowing for direct hands-on interaction. From interacting with TN, users can explore how local interactions lead to global phenomena; one of the fundamental concepts of complex systems.

Fig. 4. We have demonstrated Tangible Networks at a number of events, and reception has been overwhelmingly positive.

TN has been demonstrated at a range of events including science festivals, university open days, school lessons, summer schools, undergraduate courses and academic conferences (Fig. 4). The attractive visualisation and interactive aspect has captured the interest of a wide range of audiences ranging from young children to senior academics. Through displaying multiple models at events, we demonstrated the adaptability of network science to explain a variety of different systems, and people have been impressed by the breadth of systems studied with networks. We have used TN for demonstrating particular system behaviours, as well as letting users freely explore and interact with the models and facilitating further discussion.

We believe that TN is well suited for educational use. Alternative learning activities are used to vary teaching styles in order to address different ways of learning [13, 16, 22]. TN offers a way to open the 'black box' of in-silico simulations in order to facilitate the understanding of concepts in mathematical modelling, network science and graph theory. These models can be discussed in as much technical detail as required, making it suitable from primary schools to universities. The TN toolkit could also be used to teach programming and electronics; making a device or program that 'does something useful' is motivating and rewarding.

We would like to encourage others to use the platform as it is, or develop it further for their own work. The necessary files to make the TN hardware and software are all open source. Designs are available online, along with information about the project [25]. We have written an Arduino library for the TN hardware which facilitates further software development.

Communication of scientific ideas is of utmost importance, be it to funding bodies, schoolchildren, undergraduate students, fellow academics and members of the general public. We hope that TN will be useful in this regard.

Acknowledgements

This work has been funded by the Bristol Centre for Complexity Sciences, through EPSRC grant EP/I013717/1. EK wishes to acknowledge funding from the James Dyson Foundation.

References

1. Arduino Project: Arduino, www.arduino.cc
2. Arenas, A., Díaz-Guilera, A., Kurths, J., Moreno, Y., Zhou, C.: Synchronization in complex networks. Physics Reports 469(3), 93–153 (2008)
3. BCCS: BCCS Outreach (2014), http://www.bristol.ac.uk/bccs/public-engagement/
4. Bdeir, A.: Electronics as material: littleBits. Proc. ACM TEI '09 pp. 397–400 (2009), http://dl.acm.org/citation.cfm?id=1517743
5. Bubela, T., Nisbet, M.C., Borchelt, R., Brunger, F., Critchley, C., Einsiedel, E., Geller, G., Gupta, A., Hampel, J., Hyde-Lay, R., et al.: Science communication reconsidered. Nature Biotechnology 27(6), 514–518 (2009)
6. Buck, J.: Synchronous rhythmic flashing of fireflies. ii. Quarterly review of biology pp. 265–289 (1988)
7. Campos, P.R.A., de Oliveira, V.M., Moreira, F.G.B.: Small-world effects in the majority-vote model. Phys. Rev. E 67, 026104 (Feb 2003), http://link.aps.org/doi/10.1103/PhysRevE.67.026104
8. Dorfler, F., Bullo, F.: Synchronization and transient stability in power networks and nonuniform kuramoto oscillators. SIAM Journal on Control and Optimization 50(3), 1616–1642 (2012)
9. Filatrella, G., Nielsen, A.H., Pedersen, N.F.: Analysis of a power grid using a kuramoto-like model. The European Physical Journal B-Condensed Matter and Complex Systems 61(4), 485–491 (2008)

10. FitzHugh, R.: Impulses and physiological states in theoretical models of nerve membrane. Biophysical journal 1(6), 445–466 (1961)
11. Girouard, A., Solovey, E., Hirshfield, L.: Smart Blocks: a tangible mathematical manipulative. Proc. ACM TEI '07 pp. 183–186 (2007), http://dl.acm.org/citation.cfm?id=1227007
12. Glowacki, D.R., O'Connor, M., Calabro, G., Price, J., Tew, P., Mitchell, T., Hyde, J., Tew, D., Coughtrie, D.J., McIntosh-Smith, S.: A GPU-accelerated immersive audiovisual framework for interaction with molecular dynamics using consumer depth sensors. Faraday Discussions (2014), http://pubs.rsc.org/en/Content/ArticleLanding/2014/FD/c4fd00008k
13. Grasha, A.: Teaching with Style: A Practical Guide to Enhancing Learning by Understanding Teaching and Learning Styles. Curriculum for change series, Alliance Publishers (1996)
14. Horn, M.S., Jacob, R.K.J.: Designing Tangible Programming Languages for Classroom Use. Proc. ACM TEI '07 pp. 159–162 (2007), http://dl.acm.org/citation.cfm?id=1227003
15. Ishii, H., Ullmer, B.: Tangible bits: towards seamless interfaces between people, bits and atoms. Proc. ACM CHI '97 pp. 234–241 (1997), http://dl.acm.org/citation.cfm?id=258715
16. Kolb, D.A., et al.: Experiential learning: Experience as the source of learning and development, vol. 1. Prentice-Hall Englewood Cliffs, NJ (1984)
17. Kuramoto, Y.: Chemical oscillations, waves, and turbulence. Courier Dover Publications (2003)
18. Marshall, P.: Do tangible interfaces enhance learning? Proc. ACM TEI '07 pp. 163–170 (2007), http://dl.acm.org/citation.cfm?id=1227004
19. Mitchell, T., Hyde, J., Tew, P., Glowacki, D.: danceroom Spectroscopy: At the Frontiers of Physics, Performance, Interactive Art and Technology. Leonardo p. 140826115909005 (Aug 2014), http://www.mitpressjournals.org/doi/abs/10.1162/LEON_a_00924
20. Nagumo, J., Arimoto, S., Yoshizawa, S.: An active pulse transmission line simulating nerve axon. Proceedings of the IRE 50(10), 2061–2070 (1962)
21. Newman, M.: Networks: an introduction. Oxford University Press (2010)
22. Riechmann, S.W., Grasha, A.F.: A rational approach to developing and assessing the construct validity of a student learning style scales instrument. The Journal of Psychology 87(2), 213–223 (1974)
23. Rubenstein, M., Cornejo, a., Nagpal, R.: Programmable self-assembly in a thousand-robot swarm. Science 345(6198), 795–799 (Aug 2014), http://www.sciencemag.org/cgi/doi/10.1126/science.1254295
24. Schweikardt, E., Gross, M.: roBlocks: a robotic construction kit for mathematics and science education. Proc. ACM ICMI '06 (2006), http://dl.acm.org/citation.cfm?id=1181010
25. Tangible Networks: , www.tangiblenetworks.net
26. Wolfendale Committee and others: Wolfendale committee final report (1995)
27. Wynne, B.: Public engagement as a means of restoring public trust in science–hitting the notes, but missing the music? Public Health Genomics 9(3), 211–220 (2006)