Manuscript Title:
with Forced Linebreak[*]

Ann Author[†] and Second Author[‡]
Authors' institution and/or address
This line break forced with \\
(MUSO Collaboration)

Charlie Author[§]
Second institution and/or address
This line break forced and
Third institution, the second for Charlie Author

Delta Author
Authors' institution and/or address
This line break forced with \\
(CLEO Collaboration)

(Dated: June 19, 2022)

Because of its powerful ability to extract fundamental features from complex systems, deep learning has a wide range of applications in areas such as image recognition, speech processing, and biological systems. In particular, in the field of condensed matter physics, deep learning has demonstrated the unique ability to accurately approximate complex physical systems. In the past few years, traditional neural networks have achieved good results in condensed matter physics both in terms of feature extraction and model generation. However, the limitation of configuration size has led to the generation of only single-size configurations, which limits the generalizability of neural networks. Traditional deep learning algorithms assume that data samples are independent of each other, but in real physical configurations, each spin node is associated with other nodes and edges in the configuration. This problem is solved by the emergence of graph neural networks, which are irregular, can adapt to arbitrary size configurations, and can capture the interdependencies between different nodes. Variational self-coding is an unsupervised learning algorithm that can effectively model the distribution of training data. In this paper, we use this feature of graph neural networks in combination with variational self-coding to construct Ising model simulators, and the generated Ising models are very similar to the data simulated by traditional Monte Carlo methods. The results show that graph neural networks are a promising computational tool in the field of condensed matter physics.

I. INTRODUCTION

Over the past year, physicists have started to use deep learning techniques to study the field of condensed matter physics. Most of the tasks are done by supervised learning algorithms. In supervised learning, the algorithm is trained on data with labels, which are assigned to data points. After successful training, high precision predictions can be made for labels of previously unseen data. In addition to supervised learning, there are unsupervised learning algorithms that can find structure in unlabeled data. It has become possible to use unsupervised learning techniques to reproduce the Monte Carlo sampling state of the Ising model, and discovering phase transitions in an unsupervised manner has become mainstream. Restricted Boltzmann machines, variational self-coding, and generative adversarial networks have been widely used for the simulation of Ising models with good results. However, none of these models can take into account the molecule-molecule interactions and the influence of the edges between molecules in the physical model during training. Also, the microstructure of many physical models is irregular, and the generated models are poorly generalized and can generally only accommodate one size at one temperature. This requires a deep neural network that can consider unstructured data and can take into account node-to-node and node-to-edge relationships.

Fortunately, the research on graph neural networks in the field of deep learning has been growing in enthusiasm in recent years, and the excellent ability of graph neural networks to handle unstructured data makes them particularly useful for applications in physics, biology, and chemistry. Most physical models are essentially a graph structure. Compared with traditional convolutional networks, graph convolutional networks are better at encoding structural information of graphs and can learn better and deeper representation features. Graph vari-

[*] A footnote to the article title
[†] Also at Physics Department, XYZ University.
[‡] Second.Author@institution.edu
[§] http://www.Second.institution.edu/~Charlie.Author

ational self-coding is an unsupervised graph generation algorithm, which is the product of combining graph neural network and generative network, and it can effectively use the graph structure of data to simulate the distribution of training data.

In this paper, we integrate a graph convolutional network into a graph variational self-encoder framework to construct an Ising model simulator, which is used to simulate Ising model configurations at different temperatures. The trained simulator can effectively generate Ising model states with physical properties that are not different from those obtained from conventional Monte Carlo simulations. It can also be adapted to a wide range of different sizes and significantly reduces the simulation time. We will first introduce the general form of the Ising model, graph variational neural networks and graph convolutional networks. In the core of the paper, we construct a graph variational self-encoder combined with a graph convolutional network to simulate the 2D Ising model at different temperatures and verify the validity of our method by comparing the generated data with those from Monte Carlo simulations.

II. ISING MODEL AND GRAPH NEURAL NETWORKS

A. Ising model

In statistical physics, the Ising model is described as the set of binary spins with coupling interactions in some lattice. Considering that N spins s=s_i can take the value ± 1, where the index i marks the position of the spin s_i, the standard Hamiltonian of the Ising system includes only the nearest-neighbor interactions, and each spin direction may be "up" (+1) or "down" (-1), although the generalized model may include long-range interactions and more spin-selective directions. The standard Hamiltonian quantities are:

$$H = -J \sum_{neighbors} S_i * S_j \tag{1}$$

B. Graph neural network

A graph is a data structure that models a set of objects (nodes) and their relationships (edges). This year, research on analyzing graphs with machine learning methods has received increasing attention due to the powerful expressiveness of graph structures. Graph neural network is a general term for algorithms that use neural networks to learn graph structured data, extract and uncover features and patterns in graph structured data, and meet the needs of graph learning tasks such as clustering, classification, prediction, segmentation, and generation. One motivation for the development of graph neural networks originated from convolutional neural networks. The widespread use of convolutional neural networks has led to breakthroughs in machine learning and opened the era of deep learning. However, convolutional neural networks can only extract potential features in regular Euclidean space data, which cannot correspond well to the complex and variable graph data in reality, and how to apply convolutional neural networks to the non-Euclidean space of graph structure has become the key problem of graph neural network models. The descriptions and definitions of the relevant symbols in graph neural networks are given below

Definition 1 *Graph:*The graph is formed by the nodes and the edges connecting the nodes, usually noted as $G = (V, E)$. where $V = \{v_1, v_2, ..., v_n\}$ represents the set of nodes,$E = \{e_1, e_2, ..., e_n\}$ represents the set of edges, and edges can also be represented as (v_1, v_2). Usually nodes are also called vertices or intersections. Edges are also called links or arcs. A generic graph representation is a quintet: $G(V, E, A, X, D)$. Where A represents the adjacency matrix of the graph, X represents the identity matrix of the nodes, and D represents the degree matrix.

Definition 2 *Adjacency matrix*: The adjacency matrix of a graph refers to the matrix used to represent the connectivity of the nodes in the graph. This matrix can be binary or weighted. For an undirected graph with N nodes, the adjacency matrix is an N*N real symmetric matrix.

Definition 3 *Degree matrix*: The degree of a node represents the number of edges connected to that node. The degree matrix of a graph is the matrix that describes the degree of each node in the graph. The degree matrix is a diagonal matrix, and for undirected graphs, only the incoming degree matrix or the outgoing degree matrix is generally used.

Definition 4 *Combinatorial Laplacian matrix*, also known as standard Laplacian matrix, is a combination of diagonal matrix and adjacency matrix:

$$L = D - A \tag{2}$$

The matrix has non-zero elements only at the central node and at the nodes connected to the first order, and zero everywhere else. Laplacian matrix is also a form of graph.

Definition 5 *Normalized Laplacian matrix*

$$L^{sym} = I - D^{-1/2} A D^{-1/2} \tag{3}$$

Its element value is :

$$L^{sym}_{(i,j)} = \begin{cases} 1, & \text{i=j,}deg(v_i) \neq 0 \\ \frac{-1}{\sqrt{deg(v_i) deg(v_j)}}, & i \neq j \text{ ,}v_i \text{ and } v_j \text{ are adjacent} \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

where $deg(v)$ is denoted as the degree of node v

1. *Graph Convolutional Network*

Giving a `\label{#1}` command directly after the `\begin{subequations}`, allows you to reference all the equations in the `subequations` environment. For example, the equations in the preceding subequations environment were Eqs. (**??**).

2. *Wide equations*

The equation that follows is set in a wide format, i.e., it spans the full page. The wide format is reserved for long equations that cannot easily be set in a single column:

$$\mathcal{R}^{(\mathrm{d})} = g^e_{\sigma_2} \left(\frac{[\Gamma^Z(3,21)]_{\sigma_1}}{Q^2_{12} - M^2_W} + \frac{[\Gamma^Z(13,2)]_{\sigma_1}}{Q^2_{13} - M^2_W} \right) + x_W Q_e \left(\frac{[\Gamma^\gamma(3,21)]_{\sigma_1}}{Q^2_{12} - M^2_W} + \frac{[\Gamma^\gamma(13,2)]_{\sigma_1}}{Q^2_{13} - M^2_W} \right) . \tag{5}$$

This is typed to show how the output appears in wide format. (Incidentally, since there is no blank line between the `equation` environment above and the start of this paragraph, this paragraph is not indented.)

III. CROSS-REFERENCING

REVTEX will automatically number such things as sections, footnotes, equations, figure captions, and table captions. In order to reference them in text, use the `\label{#1}` and `\ref{#1}` commands. To reference a particular page, use the `\pageref{#1}` command.

The `\label{#1}` should appear within the section heading, within the footnote text, within the equation, or within the table or figure caption. The `\ref{#1}` command is used in text at the point where the reference is to be displayed. Some examples: Section I on page 1, Table I, and Fig. 1.

TABLE I. A table that fits into a single column of a two-column layout. Note that REVTEX 4 adjusts the intercolumn spacing so that the table fills the entire width of the column. Table captions are numbered automatically. This table illustrates left-, center-, decimal- and right-aligned columns, along with the use of the `ruledtabular` environment which sets the Scotch (double) rules above and below the alignment, per APS style.

Left[a]	Centered[b]	Decimal	Right
1	2	3.001	4
10	20	30	40
100	200	300.0	400

[a] Note a.

[b] Note b.

IV. FLOATS: FIGURES, TABLES, VIDEOS, ETC.

Figures and tables are usually allowed to “float”, which means that their placement is determined by LATEX, while the document is being typeset.

Use the `figure` environment for a figure, the `table` environment for a table. In each case, use the `\caption` command within to give the text of the figure or table caption along with the `\label` command to provide a key for referring to this figure or table. The typical content of a figure is an image of some kind; that of a table is an alignment.

Insert an image using either the `graphics` or `graphix` packages, which define the `\includegraphics{#1}` command. (The two packages differ in respect of the optional arguments used to specify the orientation, scaling, and translation of the image.) To create an alignment, use the `tabular` environment.

The best place to locate the `figure` or `table` environment is immediately following its first reference in text; this sample document illustrates this practice for Fig. 1, which shows a figure that is small enough to fit in a single column.

In exceptional cases, you will need to move the float earlier in the document, as was done with Table II:

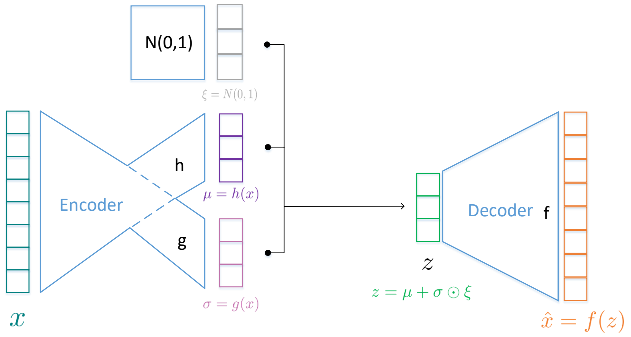

FIG. 1. A figure caption. The figure captions are automatically numbered.

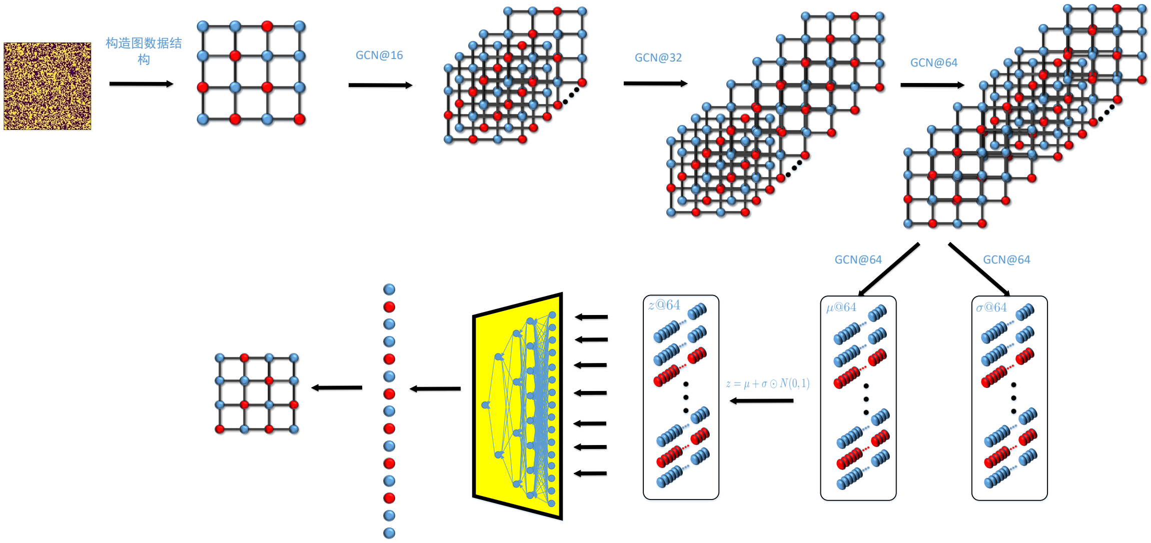

FIG. 2. Use the figure* environment to get a wide figure that spans the page in `twocolumn` formatting.

TABLE II. This is a wide table that spans the full page width in a two-column layout. It is formatted using the `table*` environment. It also demonstates the use of `\multicolumn` in rows with entries that span more than one column.

	D^1_{4h}		D^5_{4h}	
Ion	1st alternative	2nd alternative	lst alternative	2nd alternative
K	$(2e)+(2f)$	$(4i)$	$(2c)+(2d)$	$(4f)$
Mn	$(2g)$[a]	$(a)+(b)+(c)+(d)$	$(4e)$	$(2a)+(2b)$
Cl	$(a)+(b)+(c)+(d)$	$(2g)$[a]	$(4e)$[a]	
He	$(8r)$[a]	$(4j)$[a]	$(4g)$[a]	
Ag		$(4k)$[a]		$(4h)$[a]

[a] The z parameter of these positions is $z \sim \frac{1}{4}$.

LaTeX's float placement algorithms need to know about a full-page-width float earlier.

Fig. 2 has content that is too wide for a single column, so the `figure*` environment has been used.

The content of a table is typically a `tabular` environment, giving rows of type in aligned columns. Column entries separated by `&`'s, and each row ends with \\\\. The required argument for the `tabular` environment specifies how data are aligned in the columns. For instance, entries may be centered, left-justified, right-justified, aligned on a decimal point. Extra column-spacing may be be specified as well, although REVTeX 4 sets this spacing so that the columns fill the width of the table. Horizontal rules are typeset using the `\hline` command. The doubled (or Scotch) rules that appear at the top and bottom of a table can be achieved enclosing the `tabular` environment within a `ruledtabular` environment. Rows whose columns span multiple columns can be typeset using the `\multicolumn{#1}{#2}{#3}` command (for example, see the first row of Table II).

TABLE III. Numbers in columns Three–Five are aligned with the "d" column specifier (requires the `dcolumn` package). Non-numeric entries (those entries without a ".") in a "d" column are aligned on the decimal point. Use the "D" specifier for more complex layouts.

One	Two	Three	Four	Five
one	two	three	four	five
He	2	2.77234	45672.	0.69
C[a]	C[b]	12537.64	37.66345	86.37

[a] Some tables require footnotes.
[b] Some tables need more than one footnote.

Tables I, II, III, and IV show various effects. A table that fits in a single column employs the `table` environment. Table II is a wide table, set with the `table*` environment. Long tables may need to break across pages. The most straightforward way to accomplish this is to specify the `[H]` float placement on the `table` or `table*` environment. However, the LaTeX 2_ε package `longtable` allows headers and footers to be specified for each page of the table. A simple example of the use of `longtable` can be found in the file `summary.tex` that is included with the REVTeX 4 distribution.

There are two methods for setting footnotes within a table (these footnotes will be displayed directly below the table rather than at the bottom of the page or in the bib-

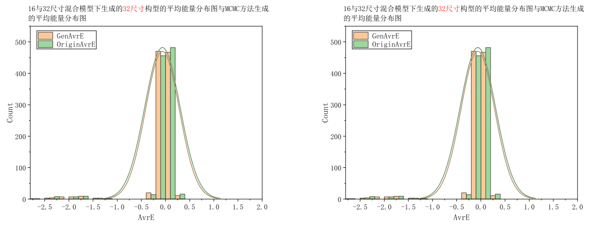

FIG. 3. name of the figure

liography). The easiest and preferred method is just to use the `\footnote{#1}` command. This will automatically enumerate the footnotes with lowercase roman letters. However, it is sometimes necessary to have multiple entries in the table share the same footnote. In this case, there is no choice but to manually create the footnotes using `\footnotemark[#1]` and `\footnotetext[#1]{#2}`. `#1` is a numeric value. Each time the same value for `#1` is used, the same mark is produced in the table. The `\footnotetext[#1]{#2}` commands are placed after the `tabular` environment. Examine the LaTeX source and output for Tables I and IV for examples.

Video **??** illustrates several features new with REVTeX4.2, starting with the `video` environment, which is in the same category with `figure` and `table`. The `\setfloatlink` command causes the title of the video to be a hyperlink to the indicated URL; it may be used with any environment that takes the `\caption` command. The `\href` command has the same significance as it does in the context of the `hyperref` package: the second argument is a piece of text to be typeset in your document; the first is its hyperlink, a URL.

Physical Review style requires that the initial citation of figures or tables be in numerical order in text, so don't cite Fig. 2 until Fig. 1 has been cited.

TABLE IV. A table with numerous columns that still fits into a single column. Here, several entries share the same footnote. Inspect the LaTeX input for this table to see exactly how it is done.

	r_c (Å)	r_0 (Å)	κr_0		r_c (Å)	r_0 (Å)	κr_0
Cu	0.800	14.10	2.550	Sn[a]	0.680	1.870	3.700
Ag	0.990	15.90	2.710	Pb[b]	0.450	1.930	3.760
Au	1.150	15.90	2.710	Ca[c]	0.750	2.170	3.560
Mg	0.490	17.60	3.200	Sr[d]	0.900	2.370	3.720
Zn	0.300	15.20	2.970	Li[b]	0.380	1.730	2.830
Cd	0.530	17.10	3.160	Na[e]	0.760	2.110	3.120
Hg	0.550	17.80	3.220	K[e]	1.120	2.620	3.480
Al	0.230	15.80	3.240	Rb[c]	1.330	2.800	3.590
Ga	0.310	16.70	3.330	Cs[d]	1.420	3.030	3.740
In	0.460	18.40	3.500	Ba[e]	0.960	2.460	3.780
Tl	0.480	18.90	3.550				

[a] Here's the first, from Ref. 10.
[b] Here's the second.
[c] Here's the third.
[d] Here's the fourth.
[e] And etc.

ACKNOWLEDGMENTS

We wish to acknowledge the support of the author community in using REVTeX, offering suggestions and encouragement, testing new versions, ...

Appendix A: Appendixes

To start the appendixes, use the `\appendix` command. This signals that all following section commands refer to appendixes instead of regular sections. Therefore, the `\appendix` command should be used only once—to setup the section commands to act as appendixes. Thereafter normal section commands are used. The heading for a section can be left empty. For example,

```
\appendix
\section{}
```

will produce an appendix heading that says "APPENDIX A" and

```
\appendix
\section{Background}
```

will produce an appendix heading that says "APPENDIX A: BACKGROUND" (note that the colon is set automatically).

If there is only one appendix, then the letter "A" should not appear. This is suppressed by using the star version of the appendix command (`\appendix*` in the place of `\appendix`).

Appendix B: A little more on appendixes

Observe that this appendix was started by using

```
\section{A little more on appendixes}
```

Note the equation number in an appendix:

$$E = mc^2. \tag{B1}$$

1. A subsection in an appendix

You can use a subsection or subsubsection in an appendix. Note the numbering: we are now in Appendix B 1.

Note the equation numbers in this appendix, produced with the subequations environment:

$$E = mc, \tag{B2a}$$
$$E = mc^2, \tag{B2b}$$
$$E \gtrsim mc^3. \tag{B2c}$$

They turn out to be Eqs. (B2a), (B2b), and (B2c).

[1] A. G. Agarwal. Proceedings of the Fifth Low Temperature Conference, Madison, WI, 1999. *Semiconductors*, 66:1238, 2001.

[2] R. Ballagh and C.M. Savage. Bose-einstein condensation: from atomic physics to quantum fluids. In C.M. Savage and M. Das, editors, *Proceedings of the 13th Physics Summer School*. World Scientific, Singapore, 2000.

[3] R. Ballagh and C.M. Savage. *Bose-Einstein condensation: from atomic physics to quantum fluids, Proceedings of the 13th Physics Summer School*. World Scientific, Singapore, 2000.

[4] E. Beutler. volume 2, chapter 7, pages 654–662. McGraw-Hill, New York, 5 edition, 1994.

[5] E. Beutler. In E. Beutler, M. A. Lichtman, B. W. Coller, and T. S. Kipps, editors, *Williams Hematology*, volume 2, chapter 7, pages 654–662. McGraw-Hill, New York, 5 edition, 1994.

[6] N. D. Birell and P. C. W. Davies. *Quantum Fields in Curved Space*. Cambridge University Press, 1982.

[7] Y. Burstyn. Proceedings of the 5th International Molecular Beam Epitaxy Conference, Santa Fe, NM. (unpublished), 5–8 October 2004.

[8] E. B. Davies and L. Parns. Trapped modes in acoustic waveguides. *Q. J. Mech. Appl. Math.*, 51:477–492, 1988.

[9] A. Einstein, Yu Podolsky, and N. Rosen. *Phys. Rev.*, 47:777, 1935.

[10] R. P. Feynman. *Phys. Rev.*, 94:262, 1954.

[11] W. K. Fields. ECE Report No. AL944, 2005. Required institution missing.

[12] Jr. G. P. Berman and Jr. F. M. Izrailev. Stability of nonlinear modes. *Physica D*, 88:445, 1983.

[13] M. P. Johnson, K. L. Miller, and K. Smith. personal communication, 1 May 2007.

[14] S. R. Kawa and S.-J. Lin. *J. Geophys. Res.*, 108(D6):4201, 2003. DOI:10.1029/2002JD002268.

[15] Donald E. Knuth. volume 1 of *The Art of Computer Programming*, section 1.2, pages 10–119. Addison-Wesley, Reading, Massachusetts, second edition, 10 January 1973. A full INBOOK entry.

[16] Donald E. Knuth. *Seminumerical Algorithms*, volume 2 of *The Art of Computer Programming*. Addison-Wesley, Reading, Massachusetts, second edition, 10 January 1981. A full BOOK entry.

[17] Jill C. Knvth. The programming of computer art. Vernier Art Center, Stanford, California, February 1988. A full BOOKLET entry.

[18] Daniel D. Lincoll. Semigroups of recurrences. In David J. Lipcoll, D. H. Lawrie, and A. H. Sameh, editors, *High Speed Computer and Algorithm Organization*, number 23 in Fast Computers, part 3, pages 179–183. Academic Press, New York, third edition, September 1977. A full INCOLLECTION entry.

[19] Larry Manmaker. *The Definitive Computer Manual*. Chips-R-Us, Silicon Valley, silver edition, April-May 1986. A full MANUAL entry.

[20] Édouard Masterly. Mastering thesis writing. Master's project, Stanford University, English Department, June-August 1988. A full MASTERSTHESIS entry.

[21] J. Nelson. U.S. Patent No. 5,693,000, 12 December 2005.

[22] J. Nelson. TWI Report 666/1999, January 1999. Required institution missing.

[23] J. K. Nelson. M.S. thesis, New York University, 1999.

[24] Automatically placing footnotes into the bibliography requires using BibTeX to compile the bibliography.

[25] Alfred V. Oaho, Jeffrey D. Ullman, and Mihalis Yannakakis. On notions of information transfer in VLSI circuits. In Wizard V. Oz and Mihalis Yannakakis, editors, *Proc. Fifteenth Annual ACM*, number 17 in All ACM Conferences, pages 133–139, New York, March 1983. ACM, Academic Press. A full INPROCEDINGS entry.

[26] W. Opechowski and R. Guccione. *Introduction to the Theory of Normal Metals*, volume IIa, page 105. Academic Press, New York, 1965.

[27] W. Opechowski and R. Guccione. Introduction to the theory of normal metals. In G. T. Rado and H. Suhl, editors, *Magnetism*, volume IIa, page 105. Academic Press, New York, 1965.

[28] W. Opechowski and R. Guccione. Introduction to the theory of normal metals. In G. T. Rado and H. Suhl, editors, *Magnetism*, volume IIa, page 105, New York, 1965. Academic Press.

[29] Wizard V. Oz and Mihalis Yannakakis, editors. *Proc. Fifteenth Annual*, number 17 in All ACM Conferences, Boston, March 1983. ACM, Academic Press. A full PROCEEDINGS entry.

[30] F. Phidias Phony-Baloney. *Fighting Fire with Fire: Festooning French Phrases*. PhD dissertation, Fanstord University, Department of French, June-August 1988. A full PHDTHESIS entry.

[31] B. Quinn, editor. *Proceedings of the 2003 Particle Accelerator Conference, Portland, OR, 12-16 May 2005*, New York, 2001. Wiley. Albeit the conference was held in 2005, it was the 2003 conference, and the proceedings were published in 2001; go figure.

[32] J. Smith. *Proc. SPIE*, 124:367, 2007. Required title is missing.

[33] J. Smith, editor. *AIP Conf. Proc.*, volume 841, 2007.

[34] J. M. Smith. In C. Brown, editor, *Molecular Dynamics*. Academic, New York, 1980.

[35] J. M. Smith. *Molecular Dynamics*. Academic, New York, 1980.

[36] J. S. Smith and G. W. Johnson. *Philos. Trans. R. Soc. London, Ser. B*, 777:1395, 2005.

[37] R. Smith. Hummingbirds are our friends. *J. Appl. Phys. (these proceedings)*, 2001. Abstract No. DA-01.

[38] S. M. Smith. Ph.D. thesis, Massachusetts Institute of Technology, 2003.

[39] V. K. Smith, K. Johnson, and M. O. Klein. Surface chemistry and preferential crystal orientation on a silicon surface. J. Appl. Phys. (submitted), 2010.

[40] W. J. Smith, T. J. Johnson, and B. G. Miller. Surface chemistry and preferential crystal orientation on a silicon surface. J. Appl. Phys. (unpublished), 2010.

[41] Tom Térrific. An $O(n \log n / \log\log n)$ sorting algorithm. Wishful Research Result 7, Fanstord University, Computer Science Department, Fanstord, California, October 1988. A full TECHREPORT entry.

[42] Ulrich Ünderwood, Ned Ñet, and Paul P̄ot. Lower bounds for wishful research results. Talk at Fanstord University (A full UNPUBLISHED entry), November, December 1988.

[43] Edward Witten, 2001.

[44] V. E. Zakharov and A. B. Shabat. Exact theory of two-dimensional self-focusing and one-dimensional self-modulation of waves in nonlinear media. *Zh. Eksp. Teor. Fiz.*, 61:118–134, 1971.

[45] Y. M. Zalkins. e-print arXiv:cond-mat/040426, 2008.