On Computing the Hamiltonian Index of Graphs

Rani M R

National Institute of Technology Calicut, India

Joint work with

Geevarghese Philip, Chennai Mathematical Institute, Chennai, India

and

Subhashini R, National Institute of Technology Calicut, India.

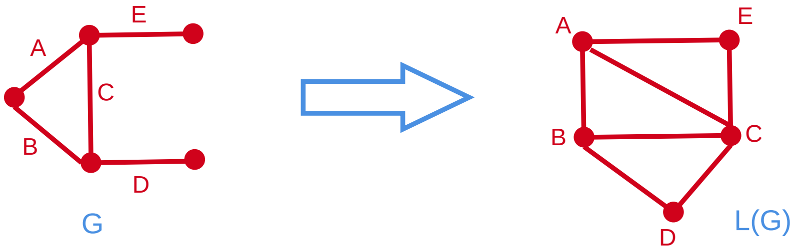
CSR 2020

Outline

Line Graph

- The line graph of an undirected graph G is another graph $L(G)$ that represents the adjacencies between edges of G.
- The vertex set of the *line graph* of a graph G—denoted $L(G)$—is the edge set $E(G)$ of G, and two vertices e, f are adjacent in $L(G)$ if and only if the edges e and f share a vertex in G.

Example

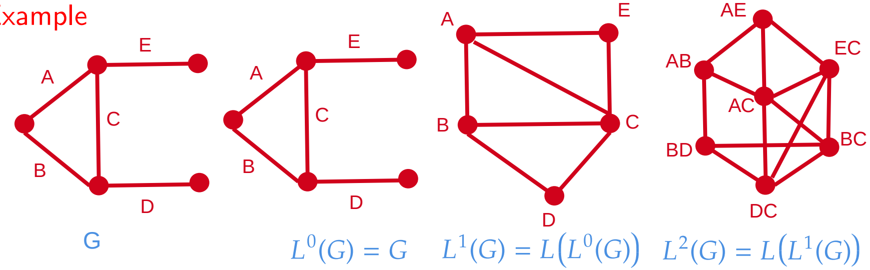


Iterated Line Graph

- For an integer $r \geq 0$ the *r- th iterated line graph* $L^r(G)$ of a graph G is defined by:
  - $L^0(G) = G$ and
  - $L^r(G) = L(L^{(r-1)}(G))$ for $r > 0$,

  where $L(G)$ denotes the line graph of G.
- If G is a connected graph which is *not* a path then $L^r(G)$ is nonempty for *all* $r \geq 0$ [CJIS90].

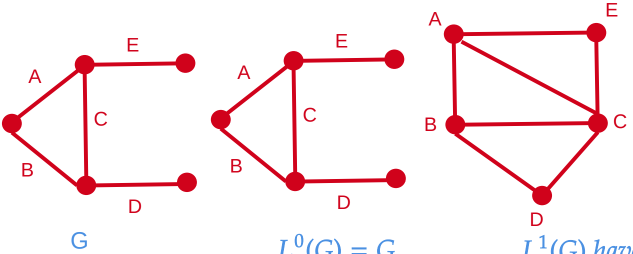
Example



Hamiltonian Index

- The *Hamiltonian Index* $h(G)$ of G is the smallest r such that $L^r(G)$ has a Hamiltonian cycle [Cha68a].

Example



$L^0(G) = G$ *does not have a HC*

$L^1(G)$ *have a HC*

$h(G) = 1$

Computing Hamiltonian Index Contd....

- Checking if $h(G) = 0$ holds $\equiv$ checking if graph G is Hamiltonian
  1. long known to be NP-complete.
  2. even when the input graph is planar and *subcubic*[GJT76].
- Checking if $h(G) = 1$ holds $\equiv$ checking if
  1. G is *not* Hamiltonian, and
  2. the line graph $L(G)$ is Hamiltonian.
  3. problem is NP-complete [Ber81] even for subcubic graphs [RWX11].

Computing Hamiltonian Index Contd....

- Checking if the line graph $L(G)$ of a graph G is Hamiltonian are equivalent to the following :
  1. G has an edge Hamiltonian cycle [Cha68a]
  2. G contains a closed trail T such that every edge in G has at least one end-point in T [HNW65]
- An *edge Hamiltonian path* of a graph G is any permutation Π of the edge set $E(G)$ of G such that every pair of consecutive edges in Π has a vertex in common.
- An *edge Hamiltonian cycle* of G is an edge Hamiltonian path of G in which the first and last edges also have a vertex in common.

Hamiltonian Index Contd....

- If the minimum degree of a graph G is at least three then $h(G) \leq 2$ holds[CW73].
- Checking whether $h(G) = t$ is NP-complete — for *any* fixed integer $t \geq 0$, even when the input graph G is subcubic.
- Goal - *Parameterized complexity analysis of the problem of computing the Hamiltonian Index.*

Parameterized Complexity

- One of the ways to deal with NP-hard problems.
- An instance of a *parameterized problem* is a pair (x, k) where x is an instance of a classical problem and k is a (usually numerical) *parameter* which captures some aspect of x.
- FPT: A problem is *fixed parameter tractable* (FPT) with respect to parameter k if there exists an algorithm running in $f(k).n^{O(1)}$ time - this running time is abbreviated as $\mathcal{O}^{\star}(f(k))$.

Problem Definition

HAMILTONIAN INDEX(HI) **Parameter:** tw

Input: A connected undirected graph $G = (V, E)$ which is not a path, a tree decomposition $\mathcal{T} = (T, \{X_t\}_{t \in V(T)})$ of G of width tw, and $r \in \mathbb{N}$.

Question: Is $h(G) \leq r$?

- **Our Result** - The HAMILTONIAN INDEX problem is fixed-parameter tractable – there is an algorithm which solves an instance $(G, \mathcal{T}, tw, r)$ of Hamiltonian Index in $\mathcal{O}^\star((1 + 2^{(\omega+3)})^{tw})$ time, where ω denotes the matrix multiplication exponent ($\omega < 2.3727$).

Related Work

- The parameterized complexity of computing $h(G)$ — has not been previously explored.
- Two special cases of checking if $h(G) \in \{0, 1\}$ — have been studied with the treewidth tw of the input graph G as the parameter
  1. Checking whether $h(G) = 0$ holds — i.e., whether G is Hamiltonian
     - was long known to be solvable in $\mathcal{O}^\star(tw^{\mathcal{O}(tw)})$ time.
     - In 2011—-showed that this can be done in randomized $\mathcal{O}^\star(4^{tw})$ time [CNP+11].
     - Bodlaender et al.[BCKN15] and Fomin et al. [FLPS16]- showed that this can be done in *deterministic* $\mathcal{O}^\star(2^{\mathcal{O}(tw)})$ time.

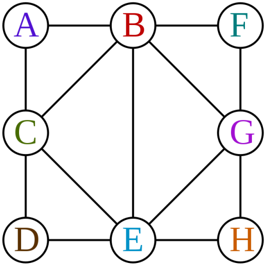
Related Work

- Checking whether $h(G) = 1$ holds — i.e. whether L(G) is hamiltonian
  - - first addressed indirectly by Lampis et al[LMMU17].
    - addressed Edge Hamiltonian Cycle(EHC) and Edge Hamiltonian Path(EHP) problems.
    - showed that EHP is FPT if and only if EHC is FPT, and that these problems can be solved in $\mathcal{O}^\star(tw^{\mathcal{O}(tw)})$ time.
  - - Misra et al. investigated an optimization variant of Edge Hamiltonian Path - called LONGEST EDGE-LINKED PATH (LELP).
    - An *edge-linked path* is a sequence of edges in which every consecutive pair has a vertex in common.
    - Given a graph G, $k \in \mathbb{N}$, and a tree decomposition $\mathcal{T}$of G of width tw as input the LELP problem asks whether G has an edge-linked path of length at least k.
    - Setting $k = |E(G)|$ yields EHP as a special case.
    - solved LELP (and hence, EHPand EHC) in $\mathcal{O}^\star((1 + 2^{(\omega+3)})^{tw})$ time.

Tree decomposition and Treewidth

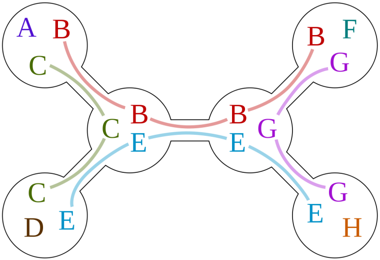
- A *tree decomposition* of a graph G is a pair $\mathcal{T} = (T, \{X_t\}_{t \in V(T)})$ where T is a tree and every vertex t of T is assigned a subset $X_t \subseteq V(G)$ of the vertex set of G.
- Each X_t - called a *bag*,

Example



A graph G

$\Longrightarrow$



A tree decomposition of G

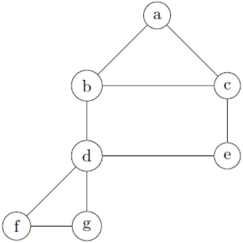
Tree decomposition and Treewidth

- The structure satisfies the following conditions:
  1. Every vertex of G is in at least one bag.
  2. For every edge uv in G there is at least one node $t \in V(T)$ such that $\{u, v\} \subseteq X_t$.
  3. For each vertex v in G the set $\{t \in V(T)\ ;\ v \in X_t\}$ of all nodes whose bags contain v, form a *connected subgraph* (i.e, a sub-tree) of T.
- The *width* of tree decomposition - maximum size of a bag, minus one.
- The *treewidth* of a graph G, denoted $tw(G)$, - minimum width of a tree decomposition of G.

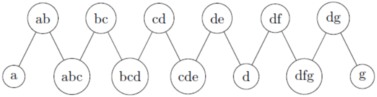
Nice Tree Decomposition

- A *nice tree decomposition* of a graph G is a tree decomposition $\mathcal{T} = (T, \{X_t\}_{t \in V(T)})$ with the following additional structure:
  1. The tree T is *rooted* at a distinguished *root* node $r \in V(T)$.
  2. The bags associated with the root node r and with every leaf node are all empty.

Example



A graph G



A nice tree decomposition of G

Nice Tree Decomposition Contd...

- Every non-leaf node is one of four types:
  1. *Introduce vertex node*: — a node $t \in V(T)$ with exactly one child node t' such that $(X_t \setminus X_{t'}) = \{v\}$ for some vertex $v \in V(G)$–i.e., *the vertex v is introduced at node t*.
  2. *Introduce edge node*: — a node $t \in V(T)$ with exactly one child node t' such that $X_t = X_{t'}$. Further, the node t is labelled with an edge $uv \in E(G)$ such that $\{u, v\} \subseteq X_t$; the edge uv is *introduced* at node t. *Every edge in the graph G is introduced at exactly one introduce edge node in the entire tree decomposition*.
  3. *Forget* node: — a node $t \in V(T)$ with exactly one child node t' such that $(X_{t'} \setminus X_t) = \{v\}$ for some vertex $v \in V(G)$; the vertex v is *forgotten* at node t.
  4. *Join* node: — a node $t \in V(T)$ with exactly two child nodes t_1, t_2 such that $X_t = X_{t_1} = X_{t_2}$.

Nice Tree Decomposition Contd.....

- For a node $t \in V(T)$ of the nice tree decomposition $\mathcal{T}$, let
  1. T_t — the subtree of T which is rooted at t.
  2. V_t — the union of all the bags associated with nodes in T_t.
  3. E_t — the set of all edges introduced in T_t, and
  4. $G_t = (V_t, E_t)$ – *the subgraph of G defined by* T_t.

Eulerian Steiner Subgraph

- A key part of our algorithm for computing $h(G)$ consists of solving Eulerian Steiner Subgraph:

EULERIAN STEINER SUBGRAPH(ESS) **Parameter:** *tw*
Input: An undirected graph $G = (V, E)$, a set of "terminal" vertice $K \subseteq V$, and a tree decomposition $\mathcal{T} = (T, \{X_t\}_{t \in V(T)})$ of G, of widt tw.
Question: Does there exist an Eulerian subgraph $G' = (V', E')$ of such that $K \subseteq V'$?

- Eulerian Steiner Subgraph problem is NP-complete[Pul79].

An FPT algorithm for Eulerian Steiner Subgraph

- Input: an instance $(G, K, \mathcal{T}, tw)$ of Eulerian Steiner Subgraph
- Our algorithm tells in $\mathcal{O}^\star((1 + 2^{(\omega+3)})^{tw})$ time whether graph G has a subgraph which is
  1. Eulerian, and
  2. contains every vertex in the terminal set K.
- Obtain a *nice* tree decomposition from $\mathcal{T}$ in polynomial time [Klo94].
- Perform dynamic programming (DP) over the bags of this nice tree decomposition.
- Pick an arbitrary terminal $v^\star \in K$ and add it to every bag of $\mathcal{T}$.
- Let $\mathcal{T}$ refer to the resulting "nearly-nice" tree decomposition in which the bags at all the leaves and the root are equal to $\{v^\star\}$.

An FPT algorithm for Eulerian Steiner Subgraph

- How an Eulerian subgraph $G' = (V', E')$ of G which contains all the terminals K interacts with the structures defined by node t in the following way
- For a bag X_t and subsets $X \subseteq X_t$, $O \subseteq X$, we say that a partition $P = \{X^1, X^2, \dots X^p\}$ of X is *valid for the combination* (t, X, O) if there exists a subgraph $G'_t = (V'_t, E'_t)$ of G_t such that
  1. $X_t \cap V(G'_t) = X$.
  2. G'_t has exactly p connected components $C_1, C_2, \dots, C_p$ and for each $i \in \{1, 2, \dots, p\}$, $X^i \subseteq V(C_i)$.
  3. Every terminal vertex from $K \cap V_t$ is in $V(G'_t)$.
  4. The set of odd-degree vertices in G'_t is exactly the set O.
- Such a subgraph G'_t of G_t - *is a witness for* $((t, X, O), P)$.

An FPT algorithm for Eulerian Steiner Subgraph

- For each node t of the tree decomposition $\mathcal{T}$ compute the set of all partitions P which are valid for the combination (t, X, O).
- At the root node r the algorithm would apply validity condition to decide the instance $(G, K, \mathcal{T}, tw)$.
- The running time of this algorithm could have a factor of tw^{tw}.
- To avoid this after computing a set $\mathcal{A}$ of valid partitions for each combination (t, X, O) we compute a representative subset $\mathcal{B} \subseteq \mathcal{A}$ and throw away the remaining partitions $\mathcal{A} \setminus \mathcal{B}$.
- Thus the number of partitions which we need to remember for any combination (t, X, O) never exceeds 2^{tw}.
- The entire DP can be done in $\mathcal{O}^{\star}((1 + 2^{(\omega+3)})^{tw})$ time.

Finding the Hamiltonian Index

- Input: An instance $(G, \mathcal{T}, tw, r)$ of Hamiltonian Index.
- Output: In $\mathcal{O}^{\star}((1+2^{(\omega+3)})^{tw})$ time outputs whether graph G has Hamiltonian Index at most r.
- If G is a connected graph on n vertices which is not a path, then $L^r(G)$ is Hamiltonian for all integers $r \geq (n-3)$ [Cha68a].
- If $r \geq (|V(G)| - 3)$ holds then our algorithm returns **yes**.
- If $r < (|V(G)| - 3)$ then it checks, for each $i = 0, 1, \ldots, r$ in increasing order, whether $h(G) = i$ holds.

Finding the Hamiltonian Index

- How we check if $h(G) = i$ holds for increasing values of i.
  1. For $i = 0$ we apply an algorithm of Bodlaender et al[BCKN15].
  2. For $i = 1$ we apply a classical result of Harary and Nash-Williams[HNW65].
  3. For checking if $h(G) \in \{2, 3\}$ holds we make use of a structural result of Hong et al. [HLTC09].
     - Let G be a connected graph with $h(G) \geq 2$ and with at least one vertex of degree at least three, and let $\tilde{H}^{(2)}, \tilde{H}^{(3)}$ be graphs constructed from G using [HLTC09]. Then
     - $h(G) = 2$ if and only if $\tilde{H}^{(2)}$ has a spanning Eulerian subgraph; and
     - $h(G) = 3$ if and only if $h(G) \neq 2$ and $\tilde{H}^{(3)}$ has a spanning Eulerian subgraph.
  4. For checking if $h(G) = i$ holds for $i \in \{4, 5, \dots\}$ we use a reduction due to Xiong and Liu [XL02].
     - If G is a connected graph with $h(G) \geq 4$, then using [XL02], we construct a graph $h'(G)$ such that $h(G) = h'(G) + 1$.

Finding the Hamiltonian Index

- Checking Hamiltonicity takes $\mathcal{O}^\star((5 + 2^{(\omega+2)/2})^{tw})$ time.
- Checking if $L(G)$ is Hamiltonian takes $\mathcal{O}^\star((1 + 2^{(\omega+3)})^{tw})$ time.
- The graphs $\tilde{H}^{(2)}$ and $\tilde{H}^{(3)}$ can each be constructed in polynomial time, and checking if each has a spanning Eulerian subgraph takes $\mathcal{O}^\star((1 + 2^{(\omega+3)})^{tw})$ time.
- The running time of the algorithm satisfies the recurrence $T(r) = \mathcal{O}^\star((1 + 2^{(\omega+3)})^{tw}) + T(r-1)$.
- The recurrence resolves to $T(r) = \mathcal{O}^\star((1 + 2^{(\omega+3)})^{tw})$.

Conclusion

- The Hamiltonian Index $h(G)$ - introduced by Chartrand in 1968.
- Checking if $h(G) = t$ holds is NP-hardfor any fixed integer $t \geq 0$.
- the problem is FPTand - $\mathcal{O}^{\star}((1 + 2^{(\omega+3)})^{tw(G)})$ time.
- This running time matches that of the current fastest algorithm, due to Misra et al. [MPS19], for checking if $h(G) = 1$ holds.
- Eulerian Steiner Subgraphproblem - $\mathcal{O}^{\star}((1 + 2^{(\omega+3)})^{tw(G)})$ time.
- Our result on Eulerian Steiner Subgraph could turn out to be useful for solving other problems as well.
- Whether there exists a matching lower bound, or can this be improved?
- Can $h(G)$ be found in the same FPT running time as it takes to check if G is Hamiltonian?

References I

[BCKN15] Hans L Bodlaender, Marek Cygan, Stefan Kratsch, and Jesper Nederlof.

Deterministic single exponential time algorithms for connectivity problems parameterized by treewidth.

Information and Computation, 243:86–111, 2015.

[BDD+16] Hans L Bodlaender, Pål Grnås Drange, Markus S Dregi, Fedor V Fomin, Daniel Lokshtanov, and Michał Pilipczuk.

A $c^k n$ 5-approximation algorithm for treewidth.

SIAM Journal on Computing, 45(2):317–378, 2016.

[Ber81] Alan A Bertossi.

The edge hamiltonian path problem is np-complete.

Information Processing Letters, 13(4-5):157–159, 1981.

[Bod98] Hans L. Bodlaender.

A partial k-arboretum of graphs with bounded treewidth.

Theoretical Computer Science, 209(1):1–45, 1998.

[BR00] R. Balakrishnan and K. Ranganathan.

A Textbook of Graph Theory.

Universitext (1979). Springer, 2000.

References II

[BR14] Mitja Bezenšek and Borut Robič.
A survey of parallel and distributed algorithms for the steiner tree problem.
Int. J. Parallel Program., 42(2):287–319, April 2014.

[BS81] Richard A. Brualdi and Robert F. Shanny.
Hamiltonian line graphs.
Journal of Graph Theory, 5(3):307–314, 1981.

[Cat88] Paul A Catlin.
A reduction method to find spanning eulerian subgraphs.
Journal of graph theory, 12(1):29–44, 1988.

[Cat92] Paul A. Catlin.
Supereulerian graphs: A survey.
Journal of Graph Theory, 16:177–196, 1992.

[CD13] Xiuzhen Cheng and Ding-Zhu Du.
Steiner trees in industry, volume 11.
Springer Science & Business Media, 2013.

References III

[CFK+15] Marek Cygan, Fedor V Fomin, Łukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michał Pilipczuk, and Saket Saurabh.

Parameterized algorithms.

Springer, 2015.

https://www.springer.com/in/book/9783319212746.

[Cha64] Gary Chartrand.

Graphs and their associated line-graphs.

PhD thesis, Michigan State University. Department of Mathematics, 1964.

[Cha68a] Gary Chartrand.

On hamiltonian line-graphs.

Transactions of the American Mathematical Society, 134(3):559–566, 1968.

[Cha68b] Gary Chartrand.

On hamiltonian line-graphs.

Transactions of the American Mathematical Society, 134(3):559–566, 1968.

https://www.jstor.org/stable/1994875?seq=1#metadata_info_tab_contents.

[CJIS90] Paul A Catlin, T N Janakiraman, Iqbalunnisa, and N Srinivasan.
Hamilton cycles and closed trails in iterated line graphs.
Journal of graph theory, 14(3):347–364, 1990.

[Cla84] Lane H. Clark.
On hamiltonian line graphs.
Journal of Graph Theory, 8(2):303–307, 1984.

[CLL+03] Zhi-Hong Chen, Hong-Jian Lai, Xiangwen Li, Deying Li, and Jingzhong Mao.
Eulerian subgraphs in 3-edge-connected graphs and hamiltonian line graphs.
Journal of Graph Theory, 42(4):308–319, 2003.

[CMP+14] Marek Cygan, Dániel Marx, Marcin Pilipczuk, Michał Pilipczuk, and Ildikó Schlotter.
Parameterized complexity of eulerian deletion problems.
Algorithmica, 68(1):41–61, 2014.
`https://link.springer.com/article/10.1007/s00453-012-9667-x`.

[CNP+11] Marek Cygan, Jesper Nederlof, Marcin Pilipczuk, Michal Pilipczuk, Joham MM van Rooij, and Jakub Onufry Wojtaszczyk.
Solving connectivity problems parameterized by treewidth in single exponential time.
In *2011 IEEE 52nd Annual Symposium on Foundations of Computer Science*, pages 150–159. IEEE, 2011.

References V

[CW73] Gary Chartrand and CURTISS E Wall.
On the hamiltonian index of a graph.
Studia Sci. Math. Hungar, 8:43–48, 1973.

[CY11] Leizhen Cai and Boting Yang.
Parameterized complexity of even/odd subgraph problems.
Journal of Discrete Algorithms, 9(3):231–240, 2011.
https://link.springer.com/chapter/10.1007/978-3-642-13073-1_9.

[DDH+14] Erik D. Demaine, Martin L. Demaine, Nicholas J.A. Harvey, Ryuhei Uehara, Takeaki Uno, and Yushi Uno.
Uno is hard, even for a single player.
Theoretical Computer Science, 521:51 – 61, 2014.

[DF13] Rodney G Downey and Michael R Fellows.
Fundamentals of Parameterized Complexity.
Springer Science & Business Media, 2013.

[DSR13] Ding-Zhu Du, JM Smith, and J Hyam Rubinstein.
Advances in Steiner trees, volume 6.
Springer Science & Business Media, 2013.

References VI

[Fle01] Herbert Fleischner.
(some of) the many uses of eulerian graphs in graph theory (plus some applications).
Discrete Mathematics, 230(1-3):23–43, 2001.

[FLPS16] Fedor V. Fomin, Daniel Lokshtanov, Fahad Panolan, and Saket Saurabh.
Efficient computation of representative families with applications in parameterized and exact algorithms.
Journal of the ACM, 63(4):29:1–29:60, 2016.

[GJS74] Michael R Garey, David S Johnson, and Larry Stockmeyer.
Some simplified np-complete problems.
In *Proceedings of the sixth annual ACM symposium on Theory of computing*, pages 47–63. ACM, 1974.

[GJT76] Michael R Garey, David S. Johnson, and R Endre Tarjan.
The planar hamiltonian circuit problem is np-complete.
SIAM Journal on Computing, 5(4):704–714, 1976.

[GMP+18] Prachi Goyal, Pranabendu Misra, Fahad Panolan, Geevarghese Philip, and Saket Saurabh.
Finding even subgraphs even faster.
Journal of Computer and System Sciences, 97:1–13, 2018.
https://doi.org/10.1016/j.jcss.2018.03.001.

References VII

[Gou81] Ronald J Gould.
On line graphs and the hamiltonian index.
Discrete Mathematics, 34(2):111–117, 1981.

[GW00] Frank Gurski and Egon Wanke.
The tree-width of clique-width bounded graphs without k n, n.
In *International Workshop on Graph-Theoretic Concepts in Computer Science*, pages 196–205. Springer, 2000.
https://link.springer.com/chapter/10.1007/3-540-40064-8_19.

[HK13] Mathias Hauptmann and Marek Karpiński.
A compendium on Steiner tree problems.
Inst. für Informatik, 2013.
http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.392.7444.

[HLTC09] Yi Hong, Jian-Liang Lin, Zhi-Sui Tao, and Zhi-Hong Chen.
The hamiltonian index of graphs.
Discrete Mathematics, 309(1):288–292, 2009.

[HNW65] Frank Harary and C St JA Nash-Williams.

On eulerian and hamiltonian graphs and line graphs.

Canadian Mathematical Bulletin, 8(6):701–709, 1965.

`https://www.cambridge.org/core/journals/canadian-mathematical-bulletin/article/on-eulerian-and-hamiltonian-graphs-and-line-graphs/96BA451CF0099F38C1B3FA867EC1A835`.

[HRWW95] Frank K Hwang, Dana S Richards, Pawel Winter, and P Widmayer.

The steiner tree problem, annals of discrete mathematics, volume 53.

ZOR-Methods and Models of Operations Research, 41(3):382, 1995.

[Klo94] T. Kloks.

Treewidth. Computations and Approximations.

Springer-Verlag LNCS, 1994.

[KT16] Frank Kammer and Torsten Tholey.

Approximate tree decompositions of planar graphs in linear time.

Theoretical Computer Science, 645:60 – 90, 2016.

[Lai88] Hong-Jian Lai.

On the hamiltonian index.

Discrete mathematics, 69(1):43–53, 1988.

References IX

[Lai98] Hong-Jian Lai.
Eulerian subgraphs containing given vertices and hamiltonian line graphs.
Discrete Mathematics, 178(1-3):93–107, 1998.

[LFW77] L Lesniak-Foster and James E Williamson.
On spanning and dominating circuits in graphs.
Canadian Mathematical Bulletin, 20(2):215–220, 1977.

[LLZ05] Dengxin Li, Hong-Jian Lai, and Mingquan Zhan.
Eulerian subgraphs and hamilton-connected line graphs.
Discrete Applied Mathematics, 145(3):422–428, 2005.

[LMMU14] Michael Lampis, Kazuhisa Makino, Valia Mitsou, and Yushi Uno.
Parameterized edge hamiltonicity.
In *Graph-Theoretic Concepts in Computer Science - 40th International Workshop, WG 2014, Nouan-le-Fuzelier, France, June 25-27, 2014. Revised Selected Papers*, volume 8747 of *Lecture Notes in Computer Science*, pages 348–359. Springer, 2014.

[LMMU17] Michael Lampis, Kazuhisa Makino, Valia Mitsou, and Yushi Uno.
Parameterized edge hamiltonicity.
Discrete Applied Mathematics, 2017.

References X

[LSY13] Hong-Jian Lai, Yehong Shao, and Huiya Yan.

An update on supereulerian graphs.

WSEAS Transactions on Mathematics, 12(9):926–940, 2013.

[LW93] Ten-Hwang Lai and Shu-Shang Wei.

The edge hamiltonian path problem is np-complete for bipartite graphs.

Information processing letters, 46(1):21–26, 1993.

[MPS19] Neeldhara Misra, Fahad Panolan, and Saket Saurabh.

On the parameterized complexity of edge-linked paths.

In *Computer Science - Theory and Applications - 14th International Computer Science Symposium in Russia, CSR 2019, Novosibirsk, Russia, July 1-5, 2019, Proceedings*, pages 286–298, 2019.

[Neb79] Ladislav Nebeskỳ.

On eulerian subgraphs of complementary graphs.

Czechoslovak Mathematical Journal, 29(2):298–302, 1979.

https://dml.cz/handle/10338.dmlcz/101606.

[NLX12] Zhaohong Niu, Hong-Jian Lai, and Liming Xiong.

Spanning subgraph with eulerian components.

Discrete Mathematics, 312(5):1013–1018, 2012.

[Pri96] Erich Prisner.
Line graphs and generalizations - a survey.
In G. Chartrand and M. Jacobson, editors, *Surveys in Graph Theory*, volume 116 of *Congressus Numerantium*, pages 193–230. Utilitas Mathematica Pub. Incorporated, 1996.

[PS12] Hans Jürgen Prömel and Angelika Steger.
The Steiner tree problem: a tour through graphs, algorithms, and complexity.
Springer Science & Business Media, 2012.

[Pul79] William R Pulleyblank.
A note on graphs spanned by eulerian graphs.
Journal of Graph Theory, 3(3):309–310, 1979.
https://onlinelibrary.wiley.com/doi/pdf/10.1002/jgt.3190030316.

[RPR85] MB Richey, R Gary Parker, and RL Rardin.
On finding spanning eulerian subgraphs.
Naval Research Logistics Quarterly, 32(3):443–455, 1985.

[RWX11] Zdeněk Ryjáček, Gerhard J Woeginger, and Liming Xiong.
Hamiltonian index is np-complete.
Discrete Applied Mathematics, 159(4):246–250, 2011.

References XII

[ST10] Ignasi Sau and Dimitrios M Thilikos.
Subexponential parameterized algorithms for degree-constrained subgraph problems on planar graphs.
Journal of Discrete Algorithms, 8(3):330–338, 2010.

[Vel83] Henk Jan Veldman.
Existence of dominating cycles and paths.
Discrete mathematics, 43(2-3):281–296, 1983.

[Wes09] Douglas B. West.
Introduction to graph theory, volume 2.
Prentice hall Upper Saddle River, 2009.

[Wil12] Virginia Vassilevska Williams.
Multiplying matrices faster than coppersmith-winograd.
In *Proceedings of the forty-fourth annual ACM symposium on Theory of computing*, pages 887–898. ACM, 2012.

[XBLL04] Liming Xiong, Haitze J Broersma, Xueliang Li, and MingChu Li.
The hamiltonian index of a graph and its branch-bonds.
Discrete mathematics, 285(1-3):279–288, 2004.

References XIII

[Xio01] Liming Xiong.
Circuits in graphs and the hamiltonian index.
Twente University Press, 2001.

[XL02] Liming Xiong and Zhanhong Liu.
Hamiltonian iterated line graphs.
Discrete Mathematics, 256(1-2):407–422, 2002.

[Zel79] Bohdan Zelinka.
Some remarks on eulerian graphs.
Czechoslovak Mathematical Journal, 29(4):564–567, 1979.
https://eudml.org/doc/13160.

[ZL11] Zhao Zhang and Hao Li.
Eulerian subgraphs containing given vertices.
SIAM Journal on Discrete Mathematics, 25(2):611–621, 2011.
https://doi.org/10.1137/060663337.