GRAPH DRAWING 2005

University of Limerick

Limerick, Ireland, 12 – 14 September 2005

GRAPH DRAWING 2005

Limerick, Ireland, 12-14 September 2005

13th International Symposium on Graph Drawing

September 12-14, 2005
Limerick, Ireland
www.gd2005.org

WELCOME

WELCOME TO LIMERICK

Fáilte is fiche chuig Luimneach. G'day and welcome to GD2005.

On behalf of the organizing committee it is our pleasure to welcome all participants to the 13^{th} International Symposium on Graph Drawing in Limerick.

During this year's conference 44 talks and 8 posters will present the very best of graph drawing research. In addition to the usual and always stimulating graph drawing contest, developers of graph drawing systems will be given a chance to showcase their software during a special session. Two eagerly anticipated talks will be given by Kurt Mehlhorn and George Robertson, two distinguished researchers in the field. Many participants will have attended already the Workshop on Network Analysis and Visualisation held on Sunday, 11^{th} September, in conjunction with GD2005.

We wish you a pleasant and productive sojourn in Limerick.

Peter Eades Patrick Healy

PROGRAM WITH ABSTRACTS

Monday, 12 September 2005

Conference Opening

08:30 – 09:00	Conference Opening

Session 1 Chair: *Patrick Healy*

09:00 – 10:00	*Kurt Mehlhorn* **Certifying Algorithms** A program computing a function f from X to Y receives an input x and returns an output y. A user must completely trust the program. He/she sees x and y and has no way to check whether it is indeed the case that $y = f(x)$. A certifying algorithm returns in addition a witness w with the property that knowledge of x, y, and w makes it easy to verify that $y = f(x)$. By "easy to verify" we mean two things. Firstly, there must be a simple program C (a checking program) that given x, y, and I checks whether indeed $y = f(x)$. The program C should be so simple that its correctness is obvious. Secondly, the running time of C on inputs x, y, and I should be no larger than the time required to compute $f(x)$ from scratch. This guarantees that the checking program C can be used without severe penalty in running time. We give a simple example, the task of deciding whether a graph is bipartite. A certifying algorithm returns a two-coloring if the input graph is bipartite and returns an odd cycle if the input graph is not bipartite. Certifying algorithms have many advantages over conventional algorithms. In particular, they can be tested on *every* input and not just on inputs for which the result is known by other means, and they are reliable in the sense that they either give the correct answer or raise an error. Most programs in the LEDA library are certifying. The design of certifying programs and certifying data stuctures raises new algorithmic questions. We give a survey of results: • certifying graph algorithms • checking data structures • certifying geometric algorithms • the advantages of certifying algorithms. • can every algorithm be made certifying?
10:00 – 10:25	*János Pach, Geza Toth* **Crossing Number of Toroidal Graphs** It is shown that if a graph of n vertices can be drawn on the torus without edge crossings and the maximum degree of its vertices is at most d, then its planar crossing number cannot exceed $c_d n$, where c_d is a constant depending only on d. This bound, conjectured by Brass, cannot be improved, apart from the value of the constant. We strengthen and generalize this result to the case when the graph has a crossing-free drawing on an orientable surface of higher genus and there is no restriction on the degrees of the vertices.
10:25 – 10:50	*Christoph Buchheim, Dietmar Ebner, Michael Jünger, Gunnar W. Klau, Petra Mutzel, Rene Weiskircher* **Exact Crossing Minimization** The crossing number of a graph is the minimum number of edge crossings in any drawing of the graph into the plane. This very basic property has been studied extensively in the literature from a theoretic point of view and many bounds exist for a variety of graph classes. In this paper, we present the first algorithm able to compute the crossing number of general sparse graphs of moderate size and present computational results on a popular benchmark set of graphs. The approach uses a new integer linear programming formulation of the problem combined with strong heuristics and problem reduction techniques. This enables us to compute the crossing number for 91 percent of all graphs on up to 40 nodes in the benchmark set within a time limit of five minutes per graph.

Session 2 Chair: *Dorothea Wagner*

11:20 – 11:45	*Seok-Hee Hong, Nikola S. Nikolov* **Hierarchical Layouts of Directed Graphs in Three Dimensions** We introduce a new graph drawing convention for 3D hierarchical drawings of directed graphs. The vertex set is partitioned into layers of vertices drawn in parallel planes. The vertex set is further partitioned into $k \geq 2$ subsets, called walls. The layout consists of a set of parallel walls which are perpendicular to the set of parallel planes of the layers. We also outline a method for computing such layouts and introduce four alternative algorithms for partitioning the vertex set into walls which address different aesthetic requirements.
11:45 – 12:10	*Emilio Di Giacomo, Giuseppe Liotta, Henk Meijer, Stephen K. Wismath* **Volume Requirements of 3D Upward Drawings** This paper studies the problem of drawing directed acyclic graphs in three dimensions in the straight-line grid model, and so that all directed edges are oriented in a common (upward) direction. We show that there exists a family of outerplanar directed acyclic graphs whose volume requirement is super-linear. We also prove that for the special case of rooted trees a linear volume upper bound is achievable.
12:10 – 12:35	*Maurizio Patrignani* **Complexity Results for Three-dimensional Orthogonal Graph Drawing** We introduce the 3SAT reduction framework which can be used to prove the NP-hardness of finding three-dimensional orthogonal drawings with specific constraints. We use it to show that finding a drawing of a graph whose edges have a fixed shape is NP-hard. Also, it is NP-hard finding a drawing of a graph with nodes at prescribed positions when a maximum of two bends per edge is allowed. We comment the impact of these results on the two open problems of determining whether a graph always admits a 3D orthogonal drawing with at most two bends per edge and of characterizing orthogonal shapes admitting a drawing without intersections.
12:35 – 12:50	*Olivier Devillers, Hazel Everett, Sylvain Lazard, Maria Pentcheva, Stephen K. Wismath* **Drawing K_n in Three Dimensions with One Bend per Edge** (Short Paper) We give a drawing of K_n in three dimensions in which vertices are placed at integer grid points and edges are drawn crossing-free with at most one bend per edge in a volume bounded by $O(n^{2.5})$. This represents a significant improvement over previous drawings in this model.

Session 3 Chair: *János Pach*

14:00 – 14:25	*Md. Saidur Rahman, Noritsugu Egi, Takao Nishizeki* **No-Bend Orthogonal Drawings of Series-Parallel Graphs** In a no-bend orthogonal drawing of a plane graph, each vertex is drawn as a point and each edge is drawn as a single horizontal or vertical line segment. A planar graph is said to have a no-bend orthogonal drawing if at least one of its plane embeddings has a no-bend orthogonal drawing. Every series-parallel graph is planar. In this paper we give a linear-time algorithm to examine whether a series-parallel graph G of the maximum degree three has a no-bend orthogonal drawing and to find one if G has.
14:25 – 14:50	*Mark de Berg, Elena Mumford, Bettina Speckmann* **On Rectilinear Duals for Vertex-Weighted Plane Graphs** Let $G = (V, E)$ be a plane triangulated graph where each vertex is assigned a positive weight. A rectilinear dual of G is a partition of a rectangle into $\|V\|$ simple rectilinear regions, one for each vertex, such that two regions are adjacent if and only if the corresponding vertices are connected by an edge in E. A rectilinear dual is called a cartogram if the area of each region is equal to the weight of the corresponding vertex. We show that every vertex-weighted plane triangulated graph G admits a cartogram of constant complexity, that is, a cartogram where the number of vertices of each region is constant.
14:50 – 15:15	*Huaming Zhang, Xin He* **An Application of Well-Orderly Trees in Graph Drawing** Well-orderly trees seems to have the potential of becoming a powerful technique capable of deriving new results in graph encoding, graph enumeration and graph generation. Our application of well-orderly trees in this paper provides new evidence to their power. We give more compact visibility representation of plane graphs using the properties of well orderly trees.
15:15 – 15:40	*Alice M. Dean, William Evans, Ellen Gethner, Joshua D. Laison, Mohammad Ali Safari, William T. Trotter* **Bar k-Visibility Graphs: Bounds on the Number of Edges, Chromatic Number, and Thickness** Let S be a set of horizontal line segments, or bars, in the plane. We say that G is a bar visibility graph, and S its bar visibility representation, if there exists a one-to-one correspondence between vertices of G and bars in S, such that there is an edge between two vertices in G if and only if there exists an unobstructed vertical line of sight between their corresponding bars. If bars are allowed to see through each other, the graphs representable in this way are precisely the interval graphs. We consider representations in which bars are allowed to see through at most k other bars. Since all bar visibility graphs are planar, we seek measurements of closeness to planarity for bar k-visibility graphs. We obtain an upper bound on the number of edges in a bar k-visibility graph. As a consequence, we obtain an upper bound of 12 on the chromatic number of bar 1-visibility graphs, and a tight upper bound of 8 on the size of the largest complete bar 1-visibility graph. We conjecture that bar 1-visibility graphs have thickness at most 2.

16:10 – 16:35	*Marco Gaertler, Dorothea Wagner* **A Hybrid Model for Drawing Dynamic and Evoling Graphs** Dynamic processes frequently occur in many applications. Visualizations of dynamically evolving data, for example as part of the data analysis, are typically restricted to a cumulative static view or an animation/sequential view. Both methods have their benefits and are often complementary in their use. In this article, we present a hybrid model that combines the two techniques. This is accomplished by 2.5D drawings which are calculated in an incremental way. The method has been evaluated on collaboration networks.
16:35 – 17:00	*Joshua Ho, Seok-Hee Hong* **Drawing Clustered Graphs in Three Dimensions** Clustered graphs are very useful model for drawing large and complex networks. This paper presents a new method for drawing clustered graphs in three dimensions. The method uses a *divide and conquer* approach, proposed in citehong. More specifically, it draws each cluster in a 2D plane to minimise occlusion and ease navigation. Then a 3D drawing of the whole graph is constructed by combining these 2D drawings. Our main contribution is to develop three linear time weighted tree drawing algorithms in three dimensions for clustered graph layout. Further, we have implemented a series of six different layouts for clustered graphs by combining three 3D tree layouts and two 2D graph layouts. The experimental results with metabolic pathways show that our method can produce a nice drawing of a clustered graph which clearly shows visual separation of the clusters, as well as highlighting the relationships between the clusters.
17:00 – 17:25	*Adel Ahmed, Tim Dwyer, Michael Forster, Xiaoyan Fu, Joshua Ho, Seok-Hee Hong, Dirk Koschützki, Colin Murray, Nikola S. Nikolov, Ronnie Taib, Alexandre Tarassov, Kai Xu* **GEOMI: GEOmetry for Maximum Insight** This paper describes the GEOMI system, a visual analysis tool for the visualisation and analysis of large and complex networks. GEOMI provides a collection of network analysis methods, graph layout algorithms and several graph navigation and interaction methods. GEOMI is a new generation of visual analysis tools combining graph visualisation techniques with network analysis methods. GEOMI is available from `http://www.cs.usyd.edu.au/~visual/valacon/`
17:25 – 17:50	*Emilio Di Giacomo, Walter Didimo, Luca Grilli, Giuseppe Liotta* **WhatsOnWeb: Using Graph Drawing to Search the Web** One of the most challenging issues in mining information from the World Wide Web is the design of systems that can present the data to the end user by clustering them into meaningful semantic categories. We envision that the analysis of the results of a Web search can significantly take advantage of advanced graph drawing techniques. In this paper we strengthen our point by describing the visual functionalities of WhatsOnWeb. WhatsOnWeb is a meta search clustering engine explicitly designed to make it possible that the user browses the Web by means of drawings of graphs whose nodes represent clusters of coherent data and whose edges describe semantic relationships between pairs of clusters. A prototype of WhatsOnWeb is available at `http://whatsonweb.diei.unipg.it/`.

Tuesday, 13 September 2005

Session 5 Chair: *Peter Eades*

09:00 – 10:00	*George G. Robertson* **Hierarchy Visualization: From Research to Practice** Hierarchy visualization has been a hot topic in the Information Visualization community for the last 15 years. A number of hierarchy visualization techniques have been invented, with each having advantages for some applications, but limitations or disadvantages for other applications. No technique has succeeded for a wide variety of applications. We continue to struggle with basic problems of high cognitive overhead (e.g., loss of context), poor fit to the data (e.g., problems of scale), and poor fit to the users task at hand (e.g., handling multiple points of focus). At the same time, information access improvements have made available to us much richer sources of information, including multiple hierarchies. In this talk, I will review what we know about hierarchy visualization, then describe our approach to visualization of multiple hierarchies with two techniques (Polyarchy Visualization and Schema Mapping), and conclude with lessons learned for basic hierarchy visualization and suggestions for future work.
10:00 – 10:25	*Michael T. Goodrich, George S. Lueker, Jonathan Z. Sun* **C-Planarity of Extrovert Clustered Graphs** A clustered graph has its vertices grouped into clusters in a hierarchical way via subset inclusion, thereby imposing a tree structure on the clustering relationship. The c-planarity problem is to determine if such a graph can be drawn in a planar way, with clusters drawn as nested regions and with each edge (drawn as a curve between vertex points) crossing the boundary of each region at most once. Unfortunately, as with the graph isomorphism problem, it is open as to whether the c-planarity problem is NP-complete or in P. In this paper, we show how to solve the c-planarity problem in polynomial time for a new class of clustered graphs, which we call *extrovert* clustered graphs. This class is quite natural (we argue that it captures many clustering relationships that are likely to arise in practice) and includes the clustered graphs tested in previous work by Dahlhaus as well as Feng, Eades, and Cohen. Interestingly, this class of graphs does not include, nor is it included by, a class studied recently by Gutwenger et al., therefore, this paper offers a alternative advancement in our understanding of the efficient drawability of clustered graphs in a planar way. Our testing algorithm runs in $O(n^3)$ time, implies an embedding algorithm with the same time complexity, and can be easily implemented.
10:25 – 10:50	*Pier Francesco Cortese, Giuseppe Di Battista, Maurizio Patrignani, Maurizio Pizzonia* **On Embedding a Cycle in a Plane Graph** Consider a planar drawing C_Γ of a planar graph G such that the vertices are drawn as small circles and the edges are drawn as thin strips. Consider a cycle c of G. Is it possible to draw c as a non-intersecting closed curve inside C_Γ, following the circles that correspond in C_Γ to the vertices of c and the strips that connect them? We show that this test can be done in polynomial time and study this problem in the framework of clustered planarity for highly non-connected clustered graphs.

Session 6 Chair: *Seok-Hee Hong*

11:20 – 11:45	*David Eppstein, Michael T. Goodrich, Jeremy Y. Meng* **Delta-confluent Drawings** We generalize the tree-confuent graphs to a broader class of graphs called delta-conuent graphs. This class of graphs and distance-hereditary graphs, a well-known class of graphs, coincide. Some results about the visualization of delta-confuent graphs are also given.
11:45 – 12:10	*Giuseppe Di Battista, Fabrizio Frati* **Small Area Drawings of Outerplanar Graphs** We show three linear time algorithms for constructing planar straight-line grid drawings of outerplanar graphs. The first and the second algorithm are for balanced outerplanar graphs. Both require linear area. The drawings produced by the first algorithm are not outerplanar while those produced by the second algorithm are. On the other hand, the first algorithm constructs drawings with better angular resolution. The third algorithm constructs outerplanar drawings of general outerplanar graphs with $O(n^{1.48})$ area. Further, we study the interplay between the area requirements of the drawings of an outerplanar graph and the area requirements of a special class of drawings of its dual tree.
12:10 – 12:35	*Therese Biedl, Anna Lubiw, Michael Spriggs* **Morphing Planar Graphs While Preserving Edge Directions** Two straight-line drawings P, Q of a graph (V, E) are called *parallel* if, for every edge $(u, v) \in E$, the vector from u to v has the same direction in both P and Q. We study problems of the form: given simple, parallel drawings P, Q does there exist a continuous transformation between them such that intermediate drawings of the transformation remain simple and parallel with P (and Q)? We prove that a transformation can always be found in the case of orthogonal drawings; however, when edges are allowed to be in one of three or more slopes the problem becomes NP-hard.
12:35 – 13:00	*Charalampos Papamanthou, Ioannis G. Tollis* **Applications of Parameterized *st*-Orientations in Graph Drawing Algorithms** Many graph drawing algorithms use *st*-numberings (*st*-orientations or bipolar orientations) as a first step. An *st*-numbering of a biconnected undirected graph defines a directed graph with no cycles, one single source s and one single sink t. As there exist exponentially many *st*-numberings that correspond to a certain undirected graph G, using different *st*-numberings in various graph drawing algorithms can result in aesthetically different drawings with different area bounds. In this paper, we present results concerning new algorithms for parameterized *st*-orientations, their impact on graph drawing algorithms and especially in visibility representations.

Session 7 Chair: *Petra Mutzel*

14:30 – 14:55	*Andreas Noack* **Energy-Based Clustering of Graphs with Nonuniform Degrees** In many real-world graphs, like social networks, hyperlink structures, and software dependency graphs, the degrees of the nodes vary widely. Finding clusters (i.e., dense subgraphs) in such graphs is of great practical interest, as these clusters may correspond to groups of friends or collaborators, semantically related documents, and cohesive software modules. Many existing clustering criteria and energy models are biased towards clustering together nodes with high degrees. In this paper, we develop a clustering criterion based on normalized cuts, and an energy model that uses edge repulsion instead of node repulsion to produce drawings that reveal clusters without this bias.
14:55 – 15:20	*Charis Papadopoulos, Constantinos Voglis* **Drawing Graphs using Modular Decomposition** In this paper we present an algorithm for drawing an undirected graph G which takes advantage of the structure of the modular decomposition tree of G. Specifically, our algorithm works by traversing the modular decomposition tree of the input graph G on n vertices and m edges, in a bottom-up fashion until it reaches the root of the tree, while at the same time intermediate drawings are computed. In order to achieve aesthetically pleasing results, we use grid and circular placement techniques, and utilize an appropriate modification of a well-known spring embedder algorithm. It turns out, that for some classes of graphs, our algorithm runs in $O(nm)$ time, while in general, the running time is bounded in terms of the processing time of the spring embedder algorithm. The result is a drawing that reveals the structure of the graph G and preserves certain aesthetic criteria.
15:20 – 15:45	*Ulrik Brandes, Daniel Fleischer, Thomas Puppe* **Dynamic Spectral Layout of Small Worlds** Spectral methods are naturally suited for dynamic graph layout, because continuous changes of edge weights yield continuous changes of the layout under very weak assumptions. We discuss some general principles for dynamic graph layout and derive a dynamic spectral layout approach for the animation of small-world models.
15:45 – 16:10	*Tim Dwyer, Yehuda Koren, Kim Marriott* **Stress Majorization with Orthogonal Ordering Constraints** The adoption of the stress-majorization method from multi-dimensional scaling into graph layout has provided an improved mathematical basis and better convergence properties for so-called "force-directed placement" techniques. In this paper we give an algorithm for augmenting such stress-majorization techniques with orthogonal ordering constraints and we demonstrate several graph-drawing applications where this class of constraints can be very useful.

Session 8 Chair: *David Eppstein*

16:40 – 17:05	*Chun-Cheng Lin, Hsu-Chun Yen* **On Balloon Drawings of Rooted Trees** Among various styles of tree drawing reported in the literature, balloon drawing enjoys a desirable feature of displaying tree structures in a rather balanced fashion. Each subtree in the balloon drawing of a tree is enclosed in a circle. The radius of each circle is proportional to the number of descendents associated with the root node of the subtree. In this paper, we investigate various issues related to balloon drawing of rooted trees from both algorithmic and practical viewpoints. First, we design an efficient algorithm to optimize angular resolution and aspect ratio for the balloon drawing of rooted unordered trees. For the case of ordered trees for which the center of the enclosing circle of a subtree need not coincide with the root of the subtree, flipping the drawing of a subtree (along the axis from the parent to the root of the subtree) might change both the aspect ratio and the angular resolution of the drawing. We show that optimizing the angular resolution as well as the aspect ratio with respect to this type of rooted ordered trees is reducible to the perfect matching problem for bipartite graphs, which is solvable in polynomial time. Aside from studying balloon drawing from an algorithmic viewpoint, we also propose a local magnetic spring model (which can be thought of as a variant of the popular force-directed strategies) for producing dynamic balloon drawings for rooted trees. In our framework, each edge is modelled by a magnetized spring, while each vertex is placed on a local polar magnetic field which does not interact with other magnetic fields. Our approach facilitates various operations, including interaction and navigation, on trees. With a slight modification to our force-directed based balloon drawing algorithm, we are able to apply our work to the drawing of galaxy systems, H-trees, and sparse graphs, which are of practical interest.
17:05 – 17:30	*Weidong Huang, Seok-Hee Hong, Peter Eades* **Layout effects on sociogram perception** This paper describes a within-subjects experiment in which we compare the relative effectiveness of five sociogram drawing conventions in communicating underlying network substance, based on user task performance and usability preference, in order to examine effects of different spatial layout formats on human sociogram perception. We also explore the impact of edge crossings, a widely accepted readability aesthetic. Subjective data were gathered based on the methodology of Purchase [2002]. Objective data were collected through an online system. We found that both edge crossings and conventions pose significant affects on user preference and task performance of finding groups, but either has little impact on the perception of actor status. On the other hand, the node positioning and angular solution might be more important in perceiving actor status. In visualising social networks, it is important to note that the techniques that are highly preferred by users do not necessarily always produce best task performance.
17:30 – 17:55	*Maurizio Pizzonia* **Minimum Depth Graph Embeddings and Quality of the Drawings: an Experimental Analysis** The depth of a planar embedding of a graph is a measure of the topological nesting of the biconnected components of the graph in that embedding. Motivated by the intuition that lower depth values lead to better drawings, previous works proposed efficient algorithms for finding embeddings with minimum depth. We present an experimental study that shows the impact of embedding depth minimization on important aesthetic criteria and relates the effectiveness of this approach with measures of how much the graph resembles a tree or a biconnected graph. In our study, we use a well known test suite of graphs obtained from real-world applications and a randomly generated one with favorable biconnectivity properties. In the experiments we consider orthogonal drawings computed using the topology-shape-metrics approach.
17:55 – 18:20	*Stefan Hachul, Michael Jünger* **An Experimental Comparison of Fast Algorithms for Drawing General Large Graphs** In the last decade several algorithms that generate straight-line drawings of general large graphs have been invented. In this paper we investigate some of these methods that are based on force-directed or algebraic approaches in terms of running time and drawing quality on a big variety of artificial and real-world graphs. Our experiments indicate that there exist significant differences in drawing qualities and running times depending on the classes of tested graphs and algorithms.

Wednesday, 14 September 2005

Session 9

Chair: *Stina Bridgeman*

09:00 – 09:25	*Carsten Gutwenger, Markus Chimani* **Non-Planar Core Reduction of Graphs** We present a reduction method that reduces a graph to a smaller core graph which behaves invariant with respect to planarity measures like crossing number, skewness, and thickness. The core reduction is based on the decomposition of a graph into its triconnected components and can be computed in linear time. It has applications in heuristic and exact optimization algorithms for the planarity measures mentioned above. Experimental results show that this strategy yields a reduction to 2/3 in average for a widely used benchmark set of graphs.
9:25 – 09:50	*Vida Dujmović, David R. Wood* **Graph Treewidth and Geometric Thickness Parameters** Consider a drawing of a graph G in the plane such that crossing edges are coloured differently. The minimum number of colours, taken over all drawings of G, is the classical graph parameter *thickness* $\theta(G)$. By restricting the edges to be straight, we obtain the *geometric thickness* $\overline{\theta}(G)$. By further restricting the vertices to be in convex position, we obtain the *book thickness* $\texttt{bt}(G)$. This paper studies the relationship between these parameters and the treewidth of G. Let $\mathcal{T}_k$ denote the class of graphs with treewidth at most k. Let $\theta(\mathcal{T}_k)$ / $\overline{\theta}(\mathcal{T}_k)$ / $\texttt{bt}(\mathcal{T}_k)$ denote the maximum thickness / geometric thickness / book thickness of a graph in $\mathcal{T}_k$. We prove that: • $\theta(\mathcal{T}_k) = \overline{\theta}(\mathcal{T}_k) = \lceil k/2 \rceil$, and • $\texttt{bt}(\mathcal{T}_k) = k$ for $k \leq 2$, and $\texttt{bt}(\mathcal{T}_k) = k+1$ for $k \geq 3$. The first result says that the lower bound for thickness can be matched by an upper bound, even in the more restrictive geometric setting. The second result disproves the conjecture of Ganley and Heath [*Discrete Appl. Math.* 2001] that $\texttt{bt}(\mathcal{T}_k) = k$ for all k. Analogous results are proved for outerthickness, arboricity, and star-arboricity.
09:50 – 10:15	*Michael J. Pelsmajer, Marcus Schaefer, Daniel Stefankovic* **Odd Crossing Number Is Not Crossing Number** The crossing number of a graph is the minimum number of edge intersections in a plane drawing of a graph, where each intersection is counted separately. If instead we count the number of pairs of edges that intersect an odd number of times, we obtain the *odd crossing number*. We show that there is a graph for which these two concepts differ, answering a well-known open question on crossing numbers. To derive the result we study drawings of maps (graphs with rotation systems).
10:15 – 10:40	*Jan Kynčl, Pavel Valtr* **On edges crossing few other edges in simple topological complete graphs** We study the existence of edges having few crossings with the other edges in drawings of the complete graph (more precisely, in simple topological complete graphs). A *topological graph* $T = (V, E)$ is a graph drawn in the plane with vertices represented by distinct points and edges represented by Jordan curves connecting the corresponding pairs of points (vertices), passing through no other vertices, and having the property that any intersetion point of two edges is either a common end-point or a point where the two edges properly cross. A topological graph is *simple*, if any two edges meet in at most one common point. Let $h = h(n)$ be the smallest integer such that every simple complete topological graph on n vertices contains an edge crossing at most h other edges. We show that $\Omega(n^{3/2}) \leq h(n) \leq O(n^2/\log^{1/4} n)$. We also show that the analogous function on other surfaces (torus, Klein bottle) grows as cn^2.

Session 10 Chair: *David Wood*

11:10 – 11:35	*Matthew Suderman* **Proper and Planar Drawings of Graphs on Three Layers** A proper k-layer planar graph, for an integer $k \geq 0$, is any graph with a planar drawing in which the vertices are drawn on k horizontal lines called layers and each edge is drawn a straight-line segment between end-vertices on adjacent layers. In this paper, we point out errors in an algorithm of Foessmeier and Kaufmann (CIAC, 1997) for recognizing proper 3-layer planar graphs, and then present a new characterization of proper 3-layer planar graphs that is partially based on their algorithm. Using the characterization, we then derive corresponding linear-time algorithms for recognizing and drawing proper 3-layer planar graphs. On the basis of our results, we predict that the approach of Foessmeier and Kaufmann will not easily generalize for drawings on four or more layers and suggest another possible approach along with some of the reasons why it may be more successful.
11:35 – 12:00	*Kazuyuki Miura, Machiko Azuma, Takao Nishizeki* **Convex Drawings of Plane Graphs of Minimum Outer Apices** In a convex drawing of a plane graph G, every facial cycle of G is drawn as a convex polygon. A polygon for the outer facial cycle is called an outer convex polygon. A necessary and sufficient condition for a plane graph G to have a convex drawing is known. However, it has not been known how many apices of an outer convex polygon are necessary for G to have a convex drawing. In this paper, we show that the minimum number of apices of an outer convex polygon necessary for G to have a convex drawing is, in effect, equal to the number of leaves in a triconnected component decomposition tree of a new graph constructed from G, and that a convex drawing of G having the minimum number of apices can be found in linear time.
12:00 – 12:25	*Walter Didimo, Francesco Giordano, Giuseppe Liotta* **Upward Spirality and Upward Planarity Testing** Let G be a digraph whose SPQR-tree does not have any R-node. The main result of this paper is a polynomial-time algorithm that tests whether G is upward planar and, if so, returns an upward planar representation of G. As an application of this result, a new FPT algorithm is presented that solves the upward planarity testing problem for general digraphs. Our results use the new notion of upward spirality that, informally speaking, is a measure of the "level of winding" that a triconnected component of G can have in an upward planar representation of G.
12:25 – 12:50	*Éric Fusy* **Transversal structures on triangulations, with application to straight-line drawing** We define and study a structure called transversal edge-partition related to triangulations without non empty triangles, which is equivalent to the regular edge labeling discovered by Kant and He. We study other properties of this structure and show that it gives rise to a new straight-line drawing algorithm for triangulations without non empty triangles, and more generally for 4-connected plane graphs with at least 4 border vertices. Taking uniformly at random such a triangulation with 4 border vertices and n vertices, the size of the grid is almost surely $\frac{n}{2} \cdot \left(1 - \frac{5}{27}\right) \times \frac{n}{2} \cdot \left(1 - \frac{5}{27}\right)$ up to fluctuations of order $\sqrt{n}$, and the half-perimeter is bounded by $n - 1$. The best previously known algorithms for straight-line drawing of such triangulations only guaranteed a grid of size $(\lceil n/2 \rceil - 1) \times \lfloor n/2 \rfloor$. The reduction-factor of $\frac{5}{27}$ can be explained thanks to a new bijection between ternary trees and triangulations of the 4-gon without non empty triangles.

Session 11 Chair: *Giuseppe Liotta*

14:20 – 14:45	*Michael Kaufmann, Imrich Vrt'o, Markus Geyer* **Two trees which are self-intersecting when drawn simultaneously** An actual topic in the graph drawing is the question how to draw two edge sets on the same vertex set, the so-called simultaneous drawing of graphs. The goal is to simultaneously find a nice drawing for both of the sets. It has been found out that only restricted classes of planar graphs can be drawn simultaneously using straight lines and without crossings within the same edge set. In this paper, we negatively answer one of the most often posted open questions namely whether any two trees with the same vertex set can be drawn simultaneously crossing-free in a straight line way.
14:45 – 15:10	*Martin Nöllenburg, Alexander Wolff* **A Mixed-Integer Program for Drawing High-Quality Metro Maps** In this paper we investigate the problem of drawing metro maps which is defined as follows. Given a planar graph G of maximum degree 8 with its embedding and vertex locations (e.g. the physical location of the tracks and stations of a metro system) and a set $\mathcal{L}$ of paths or cycles in G (e.g. metro lines) such that each edge of G belongs to at least one element of $\mathcal{L}$, draw G and $\mathcal{L}$ *nicely*. We first specify the niceness of a drawing by listing a number of *hard* and *soft* constraints. Then we present a mixed-integer program (MIP) which always finds a drawing that fulfills all hard constraints (if such a drawing exists) and optimizes a weighted sum of costs corresponding to the soft constraints. We also describe some heuristics that speed up the MIP. We have implemented both the MIP and the heuristics. We compare their output to that of previous algorithms for drawing metro maps and to official metro maps drawn by graphic designers.
15:10 – 15:35	*Tim Dwyer, Kim Marriott, Peter Stuckey* **Fast Node Overlap Removal** Most graph layout algorithms treat nodes as points. The problem of node overlap removal is to adjust the layout generated by such methods so that nodes of non-zero width and height do not overlap, yet are as close as possible to their original positions. We give an $O(n \log n)$ algorithm for achieving this assuming that that the number of nodes overlapping any single node is bounded by some constant. This method has two parts, a constraint generation algorithm which generates a linear number of "separation" constraints and an algorithm for finding a solution to these constraints "close" to the original node placement values. We also extend our constraint solving algorithm to give an active-set based algorithm which is guaranteed to find the optimal solution but which has considerably worse theoretical complexity. We compare our method with convex quadratic optimization and force-scan approaches and find that it is faster than either, gives results of better quality than force scan methods and similar quality to the quadratic optimisation approach.
15:35 – 16:00	*Michael Wybrow, Kim Marriott, Peter J. Stuckey* **Incremental Connector Routing** Most diagram editors and graph construction tools provide some form of automatic connector routing, typically providing orthogonal or poly-line connectors. Usually the editor provides an initial automatic route when the connector is created and then modifies this when the connector end-points are moved. None that we know of ensure that the route is of minimal length while avoiding other objects in the diagram. We study the problem of incrementally computing minimal length object-avoiding poly-line connector routings. Our algorithms are surprisingly fast and allow us to recalculate optimal connector routings fast enough to reroute connectors even during direct manipulation of an object's position, thus giving instant feedback to the diagram author.

Session 12 Chair: *Michael Kaufmann*

16:30 – 16:55	*Ileana Streinu* **Parallel-Redrawing Mechanisms, Pseudo-Triangulations and Kinetic Planar Graphs** We study parallel redrawing graphs: graphs embedded on moving point sets in such a way that edges maintain their slopes all throughout the motion. The configuration space of such a graph is of an oriented-projective nature, and its combinatorial structure relates to rigidity theoretic parameters of the graph. A special type of kinetic structure emerges, whose events can be analyzed combinatorially. Of particular interest are those planar graphs which maintain non-crossing edges throughout the motion. Our main result is that they are (essentially) pseudo-triangulation mechanisms. These kinetic graph structures have potential applications in morphing of more complex shapes than just simple polygons.
16:55 – 17:20	*Therese Biedl, Franz J. Brandenburg, Xiaotie Deng* **Crossings and Permutations** We investigate crossing minimization problems for a set of permutations, where a crossing expresses a disarrangement between elements. The goal is a common permutation π which minimizes the number of crossings. This is known as the Kemeny optimal aggregation problem minimizing the Kendall-tau distance. Recent interest into this problem comes from application to meta-search and spam reduction on the Web. This rank aggregation problem can be phrased as a one-sided two-layer crossing minimization problem for an edge coloured bipartite graph, where crossings are counted only for monochromatic edges. Here we introduce the max-version of the crossing minimization problem, $PCM_{max} - k$, which attempts to minimize the discrimination against any permutation. We show the NP-hardness of $PCM_{max} - k$ for $k \geq 4$ permutations, and establish a 2-approximation. For two permutations the problem is solved merely by inspecting the drawings, whereas it remains open for three permutations.
17:20 – 17:35	*Emilio Di Giacomo, Giuseppe Liotta, Francesco Trotta* **How to Embed a Path onto Two Sets of Points** (Short Paper) Let R and B be two sets of points such that the points of R are colored red and the points of B are colored blue. Let P be a path such that $\|R\|$ vertices of P are red and $\|B\|$ vertices of P are blue. We study the problem of computing a crossing-free drawing of P such that each blue vertex is represented as a point of B and each red vertex of P is represented as a point of R. We show that such a drawing can always be realized by using at most one bend per edge.
17:35 – 17:50	*Maurizio Patrignani* **On Extending a Partial Straight-Line Drawing** (Short Paper) We investigate the computational complexity of the following problem. Given a planar graph in which some vertices have already been placed in the plane, place the remaining vertices to form a planar straight-line drawing of the whole graph. We show that this extensibility problem, proposed in the 2003 "Selected Open Problems in Graph Drawing", is NP-complete.

Posters 1/2

Michael Bekos, Antonios Symvonis

BLer: A Boundary Labeller for Technical Drawings

In technical drawings and medical drawings/maps it is often common to explain certain features of the drawing by blocks of text that are arranged on its boundary. Bekos et. al. introduced the term *boundary labelling* to describe this special type of labelling and presented algorithms for automatic boundary labelling. `BLer` is a prototype tool aiming to automate the boundary labelling process. It targets the area of technical and medical drawings and incorporates implementations of boundary labelling algorithms presented in [1].

[1] M. Bekos, M. Kaufmann, A. Symvonis, and A. Wolff. Boundary labeling: Models and efficient algorithms for rectangular maps. In János Pach, editor, Proc. 12th Int. Symposium on Graph Drawing (GD‘04), Lecture Notes in Computer Science, New York, September 2004.

Mustafa Bilgic, Louis Licamele, Lise Getoor, Ben Shneiderman

D-Dupe: An Interactive Tool for Entity Resolution in Social Networks

Bibliographic and other databases often contain duplicate entries for names, cities, or other entities. This paper presents D-Dupe, an interactive visualization tool designed to help users to discover and resolve duplicate nodes in a social network represented by a node link graph. D-Dupe allows for arbitrary node similarity functions that are used to guide users to potential duplicate pairs. The graph visualization allows users to focus on nodes with similar attribute values, but more importantly, allows them to see common relationship patterns. Users can resolve the ambiguity by merging nodes, or by specifying that the nodes are in fact distinct. The entity resolution process is iterative; as pairs of nodes are merged, additional duplicates may be revealed. We illustrate the benefits of D-Dupe on a bibliographic collection with 1036 author nodes, linked by co-authorship relations. Users deal with complexity by applying powerful filtering tools and a meaningful substrate for node placement.

John M. Boyer

A New Method for Efficiently Generating Planar Graph Visibility Representations

A combinatorial planar embedding provides the cyclic order of the edges incident to each vertex in a crossing-free rendition of a planar graph in the plane. A visibility representation provides vertex and edge positioning information that could be used to create such a rendition. This paper provides a new linear-time method for generating visibility representations based in part on augmenting the combinatorial planar embedding decisions made by the new Boyer-Myrvold "edge addition" planarity method. Because it is based on geometric intuitions associated with the planar embedder, the new method requires neither a pre-computed *st*-numbering nor the planar graph dual.

Ali Civril, Malik Magdon-Ismail, Eli Bocek-Rivele

SDE: Graph Drawing Using Spectral Distance Embedding

We present a novel algorithm for drawing undirected connected graphs, by using a spectral decomposition of the distance matrix to approximate the graph theoretical distances. The main advantages of our algorithm are that it is “exact” (as opposed to iterative) and gives results that preserve symmetry and uniform node density, i.e., the drawings are aesthetically pleasing. Our approach has the benefits of fast spectral techniques but at the same time gives drawings of comparable or better quality to the much slower force directed approaches. The computational complexity of our algorithm is governed by the two main steps: distance matrix computation using an all-pairs shortest path algorithm, which is $O(|V||E|)$, and low order spectral decomposition, which is $O(|V|^2)$. The runtime for typical $20,000$ node graphs ranges from 100 to 150 seconds.

Seok-Hee Hong

MultiPlane: a New Framework for Drawing Graphs in Three Dimensions

This paper presents a new framework for drawing graphs in three dimensions. In general, the new framework uses a *divide and conquer* approach. More specifically, the framework divides a graph into a set of smaller subgraphs, and then draws each subgraph in a 2D plane. Finally, a 3D drawing of the graph is constructed by combining each plane, satisfying defined criteria. The framework is very flexible. Algorithms that follow this framework vary in computational complexity, depending on the type of graph and the optimisation criteria that are used. The resulting drawing may reduce visual complexity, occlusion and easier to navigate. We address new optimisation problems arising from the framework and provide simple approaches. Preliminary results suggest that the new framework can be useful for visual analysis of large and complex networks such as social networks and biological networks.

Posters 2/2

Bongshin Lee, Cynthia Sims Parr, Catherine Plaisant, Benjamin B. Bederson

Visualizing Directed Networks with Enhanced Tree Layouts: Can Interaction Tame Complexity?

Despite vast amounts of research, it is still difficult to produce good layouts for large graphs. Dense layout and occlusion make large graphs difficult to understand and interact with. On the other hand, tree layouts are more tractable than graph layouts and trees are easier to understand than graphs. We present a new interactive graph visualization system called TreePlus that is based on a tree-style layout. TreePlus transforms graphs into trees and shows the missing graph structure with visualization and interaction techniques. For example, TreePlus previews adjacent nodes, animates change of the tree structure, and gives visual hints about the graph structure. It enables users to start with a specific node and incrementally explore the graph. We suggest that our system is good for exploring the local structure of graphs and supports tasks that require reading node labels.

Sheung-Hung Poon

On Straightening Low-Diameter Unit Trees

A polygonal tree is a collection of edges joined into a tree structure embedded in space. In the tree, the edges are considered as rigid bars and the vertices flexible joints. We consider the problem of straightening of a tree in two or three dimensions by preserving the edge lengths and disallowing edge crossing. A tree can be *straightened* if all its edges can be moved to align along a common straight line preserving the edge lengths and disallowing edge crossings. The *diameter* of a tree is the number of edges of the longest path in the tree. In this paper, we show that trees of diameter 3 with different edge lengths in both two and three dimensions can always be straightened. We present some examples on locked trees of diameter 4 with different edge lengths in both two and three dimensions. Our main result is that a tree of diameter 4 with only unit edges in two dimensions can always be straightened in optimal number of moves. We conjecture that such a fact is true even in three dimensions. In particular, we can only solve the problem in three dimensions when there are at most 6 edges in the tree. The ultimate open question is whether a unit tree of any diameter in two or three dimensions can always be straightened.

Martin Siebenhaller, Michael Kaufmann

Mixed Upward Planarization - fast and robust

In this paper, we report on a new approach for calculating a mixed upward embedding of a mixed graph, that is a graph with directed and undirected edges. The problem is motivated by several applications in automatic graph layout especially UML diagrams. We present a fast and simple heuristic approach which combines the principal idea of the GT-heuristics for planarization with Sugiyama's approach for layered graph drawings. It provides a good quality and allows the application to considerably larger graphs as before in reasonable time. Furthermore, our new approach is robust enough that it can easily be extended to consider advanced concepts like swimlanes, hyperedges and clustering.

Software Demos 1/2

Adel Ahmed[1,2], Tim Dwyer[3], Michael Forster[1], Xiaoyan Fu[1], Joshua Ho[2], Seok-Hee Hong[1,2], Dirk Koschützki[5], Colin Murray[1,2], Nikola S. Nikolov[1,4], Ronnie Taib[1], Alexandre Tarassov[1,4], and Kai Xu[1]
1. National ICT Australia Ltd., 2. University of Sydney, 3. Monash University, 4. University of Limerick, 5. Institute of Plant Genetics and Crop Plant Research, Gatersleben.

GEOMI: GEOmetry for Maximum Insight

GEOMI is a tool for the visualisation and analysis of large and complex networks. GEOMI provides a collection of network analysis methods, graph layout algorithms and several graph navigation and interaction methods. GEOMI is a new generation of visual analysis tools combining graph visualisation techniques with network analysis methods. GEOMI is available from `http://www.cs.usyd.edu.au/>>visual/valacon/geomi/`.

Keith Andrews and Werner Putz
IICM, Graz University of Technology

HVS: A Framework for Visualising Hierarchies

Numerous techniques have been developed for visualising hierarchically structured information. This demo presents a new framework for the visualisation of hierarchies called the Hierarchical Visualisation System (HVS). HVS is a general framework implemented in Java. It provides a synchronised, multiple view environment for visualising, exploring, and managing large hierarchies.

Vladimir Batagelj and Andrej Mrvar
University of Ljubljana

Pajek

Pajek (spider, in Slovene) is a program, for Windows (32 bit), for analysis and visualization of large networks (having millions of vertices and lines). It is freely available since 1996, for noncommercial use. Pajek provides a framework to describe complex real-life networks – it supports temporal multirelational networks. Additional properties of vertices can be described using partitions, permutations and vectors. Besides this it offers a selection of efficient algorithms to identify and extract interesting smaller parts in the network.

Ulrik Brandes, Daniel Fleischer, and Thomas Puppe
University of Konstanz.

Dynamic Spectral Layout of Small Worlds

Spectral methods are naturally suited for dynamic graph layout, because moderate changes of a graph yield moderate changes of the layout under very weak assumptions. We demonstrate a dynamic spectral layout approach for the animation of small-world models.

Emilio Di Giacomo, Walter Didimo, Luca Grilli, and Giuseppe Liotta
University of Perugia

WhatsOnWeb: Using Graph Drawing to Search the Web

One of the most challenging issues in mining information from the World Wide Web is the design of systems that can present the data to the end user by clustering them into meaningful semantic categories. We envision that the analysis of the results of a Web search can significantly take advantage of advanced graph drawing techniques. In this paper we strengthen our point by describing the visual functionalities of WhatsOnWeb. WhatsOnWeb is a meta search clustering engine explicitly designed to make it possible that the user browses the Web by means of drawings of graphs whose nodes represent clusters of coherent data and whose edges describe semantic relationships between pairs of clusters. A prototype of WhatsOnWeb is available at `http://whatsonweb.diei.unipg.it/`.

Software Demos 2/2

Eric Durocher, Philippe Kaplan, Georg Sander, and Adrian Vasiliu
ILOG

ILOG JViews Diagrammer. Programming enchanced with styling and point-and-click tools

Tailoring is essential for practical applications of graph layout because of the unlimited variety of needs. Therefore, the success of a general-purpose product depends not only on the quality and speed of the algorithms, but also on the ease of configuration. Most products allow programmers to configure the algorithms using an Application Programming Interface (API). Additionally, ILOG JViews Diagrammer proposes a styling-based alternative approach, that can replace or complement the programming approach.

Uli Foessmeier, Brendan Madden, and Wendy Feng
Tom Sawyer Software

Tom Sawyer Visualization and Layout Components for Applications

Tom Sawyer Software produces high quality graph visualization, layout, and analysis solutions to application developers. Our software components have been used by companies, governments, and universities to build visualization applications in the areas of life sciences, data management, engineering design, intelligence, networking, and software engineering. This software has been growing architecturally and functionally to suit for the new advances in software, web, and technology industries.

S. Hanlon and S. K. Wismath
University of Lethbridge

GLuskap3D

GLuskap is a software package developed at the University of Lethbridge for creating and manipulating 3D graph drawings. Written in Python using OpenGL and wxPython, it supports Linux, MacOS X, and Microsoft Windows platforms. GLuskap is designed primarily for working with straight line and polyline (bent edge) 3D graph drawing layouts, including volume-bounded layouts such as the well-known Moment Curve algorithm and Morin and Wood's 1-bend drawing. The primary GLuskap interface supports interactive creation and manipulation of 3D graph drawings. The newest (unreleased) version of the package, demonstrated at GD 2005, also supports virtual reality hardware. This includes large-screen stereoscopic 3D display, as well as real-time position and orientation tracking of the user and true 3D interactive graph manipulation.

Andreas Noack, André Preußner, and Claus Lewerentz
Brandenburg University of Technology at Cottbus,

CrocoCosmos – Analysis and Visualization of Hierarchically Clustered Graphs

CrocoCosmos analyzes and visualizes abstractions of graphs, because large irregular graphs are generally incomprehensible without abstraction. It enables the user to interactively control abstraction by filtering vertices and edges, and by aggregating vertices and edges along a hierarchical clustering.

Adriaan Peeters
Ghent University

Grinvin – Graph Investigation Framework

The framework provides support for generating, drawing and investigating graphs and wants to provide a common platform for graph theory research. It has been designed with extendibility as the main goal and thus allows the simple integration of existing and/or external tools. We would like to present the abilities of the framework using its first application: a reimplementation of the Graffiti.pc application used in research on and teaching of graph theory.

Michael Wybrow
Monash University

Dunnart Diagram Editor

Dunnart is a basic diagram editor written as a test-bed for prototyping new layout tools as well as conducting usability testing and benchmarking for my PhD research. Aside from standard features you would expect to find in a diagram editor, it includes two sets of features that would be of particular interest to the Graph Drawing community. Both are easy to demonstrate.

PARTICIPANTS

Andreev, Radoslav, University of Limerick, Ireland, radoslav.andreev@ul.ie

Andrews, Keith, IICM, Graz University of Technology, Austria, kandrews@iicm.edu

Batagelj, Vladimir, University of Ljubljana, Slovenia, vladimir.batagelj@fmf.uni-lj.si

Baur, Michael, University of Cologne, Germany, baur@informatik.uni-karlsruhe.de

Bekos, Michael, University of Athens, Greece, mikebekos@math.ntua.gr

Bilgic, Mustafa, University of Maryland, USA, mbilgic@cs.umd.edu

Boyer, John, IBM Victoria Software Lab, Canada, boyerj@ca.ibm.com

Brandenburg, Franz, University of Passau, Germany, brandenb@informatik.uni-passau.de

Brandes, Ulrik, University of Konstanz, Germany, ulrik.brandes@uni-konstanz.de

Bridgeman, Stina, Hobart & William Smith Colleges, USA, bridgeman@hws.edu

Cahill, Tony, University of Limerick, Ireland, tony.cahill@ul.ie

Chimani, Marcus, University of Dortmund, Germany, marcus.chimani@cs.uni-dortmund.de

Civril, Ali, Rensselaer Polytechnic Institute, USA, alicivril@yahoo.com

Cortese, Pier Francesco, University of Rome III, Italy cortese@dia.uniroma3.it

Di Battista, Giuseppe, University of Rome III, Italy, gdb@dia.uniroma3.it

Di Giacomo, Emilio, University of Perugia, Italy, digiacomo@diei.unipg.it

Didimo, Walter, University of Perugia, Italy, didimo@diei.unipg.it

Dujmović, Vida, Carleton University, Canada, vida@scs.carleton.ca

Duncan, Christian, University of Miami, USA, duncan@cs.miami.edu

Dwyer, Tim, Monash University, Australia, Tim.Dwyer@infotech.monash.edu.au

Eades, Peter, The National ICT Australia Ltd., Australia, peter.eades@nicta.com.au

Eppstein, David, University of California, USA, eppstein@ics.uci.edu

Fekete, Jean Daniel, INRIA, France, fekete@lri.fr

Fleischer, Daniel, University of Konstanz, Germany, Daniel.Fleischer@inf.uni-konstanz.de

Forster, Michael, The National ICT Australia Ltd., Australia, michael.forster@nicta.com.au

Frati, Fabrizio, University of Rome III, Italy, fabriziofrati@tiscali.it

Freivalds, Karlis, University of Latvia, Latvia, Karlis.Freivalds@mii.lu.lv

Fusy, Éric, Technische University of Berlin, Germany, fusy@math.tu-berlin.de

Gaertler, Marco, University of Karlsruhe, Germany, gaertler@informatik.uni-karlsruhe.de

Ganser, Emden, AT&T Labs - Research, USA, erg@research.att.com

Geyer, Markus, University of Tuebingen, Germany, geyer@informatik.uni-tuebingen.de

Giordano, Francesco, University of Perugia, Italy, francesco.giordano@diei.unipg.it

Goerg, Carsten, Georgia Institute of Technology, USA, goerg@cc.gatech.edu

Goodrich, Michael, University of California, USA, goodrich@acm.org

Grilli, Luca, University of Perugia, Italy, luca.grilli@diei.unipg.it

Gutwenger, Carsten, University of Dortmund, Germany, carsten.gutwenger@cs.uni-dortmund.de

Hachul, Stefan, University of Cologne, Germany, hachul@informatik.uni-koeln.de

Hanlon, Sebastian, University of Lethbridge, Canada, sebastian.hanlon@uleth.ca

Harrigan, Martin, University of Limerick, Ireland, martin.harrigan@ul.ie

He, Xin, University at Buffalo, USA, xinhe@cse.buffalo.edu

Healy, Patrick, University of Limerick, Ireland, patrick.healy@ul.ie

Henry, Nathelie, INRIA Futurs, France, nathalie.henry@lri.fr

Ho, Joshua, University of Sydney, Australia, joshua.ho@student.usyd.edu.au

Hoejgaard, Jesper Just, Denmark, jhoejgaard@gmail.com

Hong, Seok-Hee, The National ICT Australia Ltd., Australia, barbara.munday@nicta.com.au

Huang, Weidong, University of Sydney, Australia, whua5569@usyd.edu.au

Juenger, Michael, University of Cologne, Germany, mjuenger@informatik.uni-koeln.de

Kaufmann, Michael, University of Tuebingen, Germany, mk@informatik.uni-tuebingen.de

Kitson, Nuala, University of Limerick, Ireland, nuala.kitson@ul.ie

Klein, Karsten, University of Dortmund, Germany, karsten.klein@cs.uni-dortmund.de

Kobourov, Stephen, University of Arizona, USA, kobourov@cs.arizona.edu

Koren, Yehuda, AT&T Labs - Research, USA, yehuda@att.com

Koschützki, Dirk, IPK Gatersleben, Germany, koschuet@ipk-gatersleben.de

Kratochvil, Jan, Charles University, Czech Republic, honza@kam.mff.cuni.cz

Kynčl, Jan, Charles University, Czech Republic, kyncl@kam.mff.cuni.cz

Laison, Josh, Colorado College, USA, jlaison@coloradocollege.edu

Lee, Bongshin, University of Maryland, USA, bongshin@cs.umd.edu

Lin, Chun-Cheng, National Taiwan University, Taiwan, sanlin@cobra.ee.ntu.edu.tw

Liotta, Giuseppe, University of Perugia, Italy, liotta@diei.unipg.it

Lynch, Karol, University of Limerick, Ireland, karol.lynch@ul.ie

Madden, Brendan, Tom Sawyer Software, USA, bmadden@tomsawyer.com

Manning, Joseph, University College Cork, Ireland, manning@cs.ucc.ie

Mehlhorn, Kurt, Max-Planck-Institut Computer Science, Germany, mehlhorn@mpi-sb.mpg.de

Miura, Kazuyuki, Fukushima University, Japan, miura@sss.fukushima-u.ac.jp

Mumford, Elena, TU Eindhoven, The Netherlands, e.mumford@tue.nl

Mutzel, Petra, University of Dortmund, Germany, petra.mutzel@cs.uni-dortmund.de

Nikolov, Nikola, University of Limerick, Ireland, nikola.nikolov@ul.ie

Noack, Andreas, University of Cottbus, Germany, an@informatik.tu-cottbus.de

Noellenburg, Martin, University of Karlsruhe, Germany, noelle@ira.uka.de

Pach, János, City College and Courant Institute, USA, pach@cims.nyu.edu

Papadopoulos, Charis, University of Ioannina, Greece, charis@cs.uoi.gr

Patrignani, Maurizio, University of Rome III, Italy, patrigna@dia.uniroma3.it

Peeters, Adriaan, Ghent University, Belgium, Adriaan.Peeters@UGent.be

Pentcheva, Maria, LORIA, France, maria.pentcheva@loria.fr

Percan, Merijam, University of Cologne, Germany, percan@informatik.uni-koeln.de

Pizzonia, Maurizio, University of Rome III, Italy, pizzonia@dia.uniroma3.it

Poon, Sheung-Hung, TU Eindhoven, The Netherlands, spoon@win.tue.nl

Quigley, Aaron, University College Dublin, Ireland, aquigley@ucd.ie

Rextin, Aimal, University of Limerick, Ireland, aimal.tariq@ul.ie

Robertson, George, Microsoft Research, USA, ggr@microsoft.com

Rodriguez, Andres, SRI International, USA, acr@ai.sri.com

Schreiber, Falk, IPK Gatersleben, Germany, schreibe@ipk-gatersleben.de

Siebenhaller, Martin, University of Tuebingen, Germany, siebenha@informatik.uni-tuebingen.de

Speckmann, Bettina, TU Eindhoven, The Netherlands, speckman@win.tue.nl

Schulz, Michael, University of Cologne, Germany, schulz@informatik.uni-koeln.de

Spriggs, Michael, University of Waterloo, Canada, mjspriggs@uwaterloo.ca

Stefankovic, Daniel, University of Rochester, USA, Daniel.Stefankovic@gmail.com

Streinu, Ileana, Smith College, USA, streinu@cs.smith.edu

Suderman, Matthew, McGill University, Canada, msuder@cs.mcgill.ca

Swift, Gemma, University of Limerick, Ireland, gemma.swift@ul.ie

Symvonis, Antonis, National Technical University of Athens, Greece, symvonis@math.ntua.gr

Tamassia, Roberto, Brown University, USA, rt@cs.brown.edu

Tarassov, Alexandre, University of Limerick, Ireland, alexandre.tarassov@ul.ie

Tollis, Ioannis, ICS-FORTH and University of Crete, Greece, tollis@ics.forth.gr

Trotta, Francesco, University of Perugia, Italy, francesco.trotta@diei.unipg.it

Valtr, Pavel, Charles University, Czech Republic, valtr@kam.mff.cuni.cz

Vasiliu, Adrian, ILOG, France, ydegbey@ilog.fr

Voglis, Constantinos, University of Ioannina, Greece, voglis@cs.uoi.gr

Vrt'o, Imrich, Institute of Mathematics, Slovakia, vrto@savba.sk

Wagner, Dorothea, University of Karlsruhe, Germany, dwagner@ira.uka.de

Whitesides, Sue, McGill University, Canada, sue@cs.mcgill.ca

Wismath, Stephen, University of Lethbridge, Canada, wismath@cs.uleth.ca

Wolff, Alexander, University of Karlsruhe, Germany, awolff@ira.uka.de

Wood, David, Universitat Politecnica de Catalunya, Spain, wood@cs.mcgill.ca

Wybrow, Michael, Monash University, Australia, mwybrow@csse.monash.edu.au

Yen, Hsu-Chun, National Taiwan University, Taiwan, yen@cc.ee.ntu.edu.tw

Zickfeld, Florian, TU Berlin, Germany, zickfeld@math.tu-berlin.de

Benefactor

sfi
science foundation ireland

Gold Sponsors

Microsoft®

Tom Sawyer
SOFTWARE®

Silver Sponsors

AbsInt
Angewandte Informatik

ENTERPRISE
IRELAND

Fáilte Ireland

ILOG
Changing the rules of business

intel®

NATIONAL
ICT AUSTRALIA
LIMITED

Contributors

DELL™

JAMESON®
IRISH WHISKEY

Lucent Technologies
Bell Labs Innovations