

# **Nic Melder PhD, BSc (Hons), MIET**

1113-191 Sherbourne Street, Toronto, Ontario, M5A 3X1, Canada

Telephone (mobile): +1 647 918 4737

Email: [linkedin@riviera76.fastmail.fm](mailto:linkedin@riviera76.fastmail.fm), Web page: <http://cv.nicmelder.fastmail.fm>

## **Key Skills/Experience**

### **Key Experience**

- 14+ years experience developing AI systems for racing games.
- 2 years experience developing physics systems for racing games (including vehicle dynamics and collision detection and response).
- 9+ years experience leading a multi-disciplinary team that was comprised of AI, physics, animation and systems programmers numbering 7 people at its peak.
- Experience working on large commercial projects with 100+ programmers and budgets in excess of £50 million (10 million+ lines of code).
- Full project life cycle experience from design to release including post release support.

### **Technical Skills**

- C++ and object oriented design and programming for PC, Xbox and Playstation.
- Experience in profiling and optimising for memory and CPU including SIMD vectorisation.
- Tools development using C#, WPF, MFC and MAXScript for multi-threaded, distributed automation applications, general tools and 3D Studio Max tools.

### **Other**

- 14 published games (2 at Ubisoft, 10 at Codemasters, 2 at Asylum Entertainment).
- Five published articles related to artificial intelligence in racing games ('Game AI Pro: Collected Wisdom of Game AI Professionals', September 2013).
- Invited speaker at the Game/AI Conference, Vienna, 2012, talking about racing AI.
- Six academic papers and thesis published in the field of VR and Haptics.
- Presented at the three leading Haptics conferences and demonstrated my Multi-finger Haptic system at various public events / conferences (including at the Science Museum in London).
- Strong analytical and design skills.
- Strong technical writing skills including academic writing.
- Experience with presenting including lecturing, conference presentations and demonstrations.
- Basic modelling with 3D Studio Max.

## **Employment History**

### **Vehicle AI Programmer (Lead) (Level 4 / Principal)**

*Ubisoft Toronto, 224 Wallace Ave, Toronto, ON M6H 1V7, Canada*

*June 2016 - Present*

As the technical and team lead for the vehicle AI team on Watchdogs: Legion, my primary responsibilities were to design, develop and direct the development of the new vehicle AI (traffic and felony/chase) systems for Watchdogs: Legion. I was also responsible for updating and enhancing the road network data systems and exporters and converting it from being a US based system (mainly grid based, driving on the right) to a UK based system (driving on the left, roundabouts, much more complex intersections).

I also managed a team of two other programmers and was responsible for planning and documentation on top of actually implementing and delivering the new systems. As the lead, I was the primary point of contact for all things vehicle AI and regularly updated the senior project leadership (lead programmers, creative and game directors, and senior production) on the current state and progress of the vehicle AI systems.

I have also spent time assisting Watchdogs 2, and Far Cry 6 with vehicle AI issues whilst their projects were being finalised for release.

Some of the notable features of the new vehicle AI system that I designed and developed follows:

**Single, unified vehicle AI system** - It was decided to replace the multiple vehicle AI systems from Watchdogs 2 with a single unified architecture. Watchdogs 2 had two completely separate systems for normal traffic and police driving / chases, a different system for boat traffic, and a fourth system for the helicopters. The new system was used for traffic (vehicle and boats), the high speed chases and also for the new drone system. This simplified the code by removing around 300 files and also meant that if a vehicle's handling or characteristics changed then it would only need to be changed in a single place in the code to make it drive correctly. The high speed chases, would use the exact same code as the normal traffic, except that it would employ additional driving behaviours (overtake) and driving characteristics (greater desired acceleration / braking, no road speed limits and could ignore traffic lights and pedestrians). A high speed felony vehicle was also much more processor performant i.e. a single high speed vehicle in the new system took under 100us whereas in the old system, it took more than 800us.

**Thread friendly and Engine agnostic** - The new Vehicle AI system was designed to be as engine agnostic as possible. Access to the systems were through well defined interfaces so that it should be relatively straightforward to transplant this system to a new engine as only the external interfaces would need to be re-implemented. Each vehicle would also internally cache all the data it needed in order to determine it's behaviour using the external interfaces. This had the added advantage that they could then be processed independently of the other systems and on a different thread if required.

**Upgraded road network system** - The road network represents the roads / sidewalks and how they connect together. Watchdogs 1 and 2 were both set in the US and were designed for a grid based road layout and driving on the right. Watchdogs: Legion required that the vehicles drove on the left and included different junction types (4 lane roundabouts, multiple types of pedestrian crossings) and much more complicated intersections (6 way traffic light intersections, 5 way stop intersections, complex forks and merges (e.g. a 6 lane road forking/merging into 3 separate lanes)). The final upgraded system now supports these exotic junction types and now supports both left and right hand driving systems.

### **Principal Programmer: Artificial Intelligence and Physics, Group Lead**

*Codemasters Software Company Ltd, Southam, Warwickshire, CV47 2DL, England  
October 2006 – February 2015*

My primary responsibilities were to design, develop and enhance the AI and Physics systems for use in the Racing Studio's critically acclaimed AAA racing titles for PC and consoles. Across the projects that I have worked on (DiRT, GRID, DiRT2, F1 2010, DiRT 3, DiRT Showdown, GRID 2, GRID:Autosport, DiRT:Rally and F1 2015), I have been responsible for planning, documenting and implementing any new AI features required. From May 2012, I was also directly responsible for implementing any new Physics systems. This also included the creation, development and maintenance of the AI and Physics specific tools and build chains using C#, WPF, MFC and MAXScript. I was also the primary point of contact between the level and car handling design teams (17 people) and the programming team as well as answering to senior management (lead programmers, chief designers and executive producers) with regard to anything related to the AI or Physics in the games.

Since 2013, a large part of my responsibilities included working with the F1 team to design and implement the next gen (EGO3) AI and Physics systems such that a common code base could be shared between the F1 studio and the Racing studio.

Some recent notable features that I have been responsible for follow. In all cases I was responsible for both developing and managing any other programmers that were involved, as well as presenting the results to senior management for sign off.

**Breakable Objects (F1 2015/EGO3 Engine)** – As the sole programmer, I worked closely with the artists to design and implement a completely new system for creating objects that would deform and / or break apart when collided with for Codemasters' next gen engine (EGO3). This required a complete redesign of how the objects were to be rigged in 3D Studio Max, a set of MAXScript tools for rigging and converting the legacy assets to the new format, writing a new build pipeline for all the physics objects (in C#) as well as the game code for activating, hiding and resetting the collidable objects (in C++). The new system had approximately 90% of the functionality that the previous system had (the 10% missing wasn't required for F1) yet was more flexible and much simpler to understand by the artists and programmers (the artist setup guide for the new system was 18 pages (including examples) vs 80 pages). Due to the simplicity of the new breakables system, the Racing Studio is planning on adopting it for use in its next title.

**Traction Control Update (GRID Autosport)** – Due to complaints from the audio department about how unnatural the cars were sounding whilst traction control was enabled, I worked with the car handling designers and audio designers to rewrite the traction control system. Since the car handling had already been signed off, it was imperative that the car handling was not affected in any detrimental way or in a way that would require that the cars were reworked. The result was that the cars sounded correct pulling away from the start line (the engine revs would dip and hold instead of redlining) and it actually improved the user experience whilst driving through the corners.

**Car Handling Upgrades (GRID 2)** – Working closely with the car handling designers, the design team and the front end programmers, a WPF tool was created for the designers to create different upgrade packs that could then be applied at runtime in game. This involved me writing the tool, rewriting the car loader to overwrite the default car handling parameters with the upgraded values, developing a 'live link' solution to allow the upgrades to be applied in real-time, building automated tools for generating the car statistics data and modifying the build system to pull data from multiple places to generate the data for the front end. Due to the flexibility of the tools and implementation, the same system was used for setting up the AI stances and as a quick car handling prototyping tool.

**Arena Based AI (DiRT Showdown)** – With only 5 months for development, working with the level and game designers and another AI programmer, this involved developing an entirely new AI system in order for the AI to actively target / avoid vehicles and fixed objects in open world arenas for multiple destruction derby type game modes. This required automatic spline generation, fixed and dynamic obstacle avoidance, a new behaviour system, a high level AI director for global decision making (who should target who, when to run away, how many to attack the player etc.) and a realistic difficulty system where cars would purposely miss the player / each other instead of just applying score / damage multipliers.

**Track Simulator Tool (post DiRT 2)** – I worked with the technical level designer to develop the track simulator tool which simulated a car driving around the track based solely upon a spline. Using data that described how fast an in game car could accelerate, brake and corner, the speed of the car at any point along the spline could be shown as well as the time to complete a lap. This tool allowed the level designers to better design their tracks to achieve a target time (e.g. a 60 second lap or 3 minute point to point) as well as create easy, difficult, slow or fast sections for tracks. In use, the simulated lap time was accurate to within five percent of the final AI time (using non-optimal racing splines) and, according to the lead level designer, this tool saved approximately 3 days per track at the design stage. It also removed the need for any expensive track shortening if the finished / modelled track was outside the desired lap time (before this tool, a track expected to last 3 minutes actually took 4.5 minutes

to drive once it had been fully modelled and delivered. It took a single level designer 2 months to rework the track down to 3 minutes without losing the feature corners / track's character / etc.).

**Equation Fitting for AI Vehicle Statistics (DiRT 2)** – In order to improve the accuracy of the vehicle's braking distance prediction, the generated speed-distance pairs were replaced with an equation where the coefficients were calculated using the Levenberg-Marquardt Least-Squares algorithm. Not only did this improve the accuracy of the braking data but it also drastically reduced the data requirements. Previously, a car with a top speed of 150mph would require 14 data pairs per surface type per grip level (DiRT had approx 20 driveable surface types) but after mapping, only 5 floats were required to describe the same surface per grip level. Mappings were also created for the vehicles acceleration, rolling and cornering profiles. Due to the reduced data requirements, braking distances were also calculated at differing brake pressures which helped the AI to drive smoother, especially at lower difficulties. Each vehicle also was able to have data unique to it instead of its vehicle class as was previously done.

**Forking Routes (DiRT 2)** – Working closely with the level designers, the ability to add multiple routes to a track was added. This required a new set of MAXScript tools as well as various tools to allow them to test the different routes. The in game AI was also enhanced to include a decision mechanism to determine the appropriate fork to take.

**Staggered Starts (DiRT 2)** – Working with the designers and the gameplay and network programmers, a system was developed that would allow individual cars to start a rally at preset time intervals. This required that a game side system was tightly integrated with the AI to ensure that the AI cars were placed forward of the start line correctly, that the cars drove up to the start line appropriately and that they set off at the correct time. Offline tools were also developed that generated all the placement data and automatically checked it into Perforce. The final system was able to drive a car from a standing start 10 metres to the start line and stop it within 5 millimetres of the target in under 5 seconds. The placement data generated and the difficulty scaling algorithm developed proved to be so accurate that it replaced the virtual times system (for generating times for AI cars still racing after the player finishes the race) and was subsequently used in many game systems where comparisons between track speeds were needed (e.g. the visual racing line in Dirt 3, determining the time between cars in GRID2).

**AI Logical Controller System (Continuous)** – I enhanced the original simplistic steering controller to include a more advanced controller with a predictive and adaptive element that made the AI cars stick much closer to their desired racing line. The pedal controller was also improved to take into account the curvature of the track ahead to determine the desired speed for that corner. The system also incorporated a 'drift' mode that allowed the cars to slide sideways around corners whilst still being fully controllable by the AI.

## **Research Fellow in Multi-Point Haptics**

*Department of Cybernetics, University of Reading, Reading, RG6 6AY, England  
October 2001 - October 2006 (Part-time until April 2002)*

**Virtual Shopping (October 2005 - October 2006)** - A virtual supermarket was created in a fully immersive 3D environment (CAVE) where the user navigates the shop through a bespoke haptic device designed to behave like a shopping trolley along with a glove interface to allow items to be picked up and manipulated. My responsibilities on the project concerned the interface between the current shelf layout software and the virtual environment, ensuring / devising mechanisms to allow a store, complete with shelves fully stocked with products, to be rendered and the physics required for the manipulation of individual items.

**Multi-Point Haptics (October 2001 - September 2005)** - Using three modified PHANTOM 1.5 haptic devices, a 3D visual, audio and haptic virtual environment was developed with a dynamic physics and collision system in order to develop haptic rendering algorithms for multi-

point haptic manipulation of virtual objects. Algorithms developed include the Friction Cone Algorithm and extensions, which allows efficient modelling of static, dynamic and viscous friction and the Residual Force Algorithms for object translation and rotation. Collision and collision response methods were also developed for when a haptic contacted object collides with another object. Other work included the incorporation of haptic feedback into DIVE (Distributed Interactive Virtual Environment) and a CAVE as well as the creation of a suite of programs for performing different psychophysical tests involving multi-finger haptics. This research was rated at International Level for all aspects by the EPSRC.

Some videos of my work can be found here:

- 2 finger manipulation of a sphere
- 2 finger manipulation of a cube
- 2 finger manipulation (early version)
- 3 finger manipulation of a sphere (early version)
- A virtual size weight illusion
- A virtual Klein bottle

Other responsibilities included lecturing, project supervision and demonstrations for prospective students as well as visiting academics and business groups.

### **AI Programmer**

*Asylum Entertainment Ltd, London, England*

*September 2000 - April 2002 (Part-time from Oct 2001)*

Developed the AI for three games (two dinosaur simulation titles for the BBC and a 3rd person action horror title). As the sole AI programmer in a small team I was responsible for the development of the predator-prey and herding AI system for the dinosaur games and the development of the finite state machines and path finding algorithms for the horror game. I also developed the animation scripting system.

### **Computer Analyst/Programmer**

*Sunray Communications, Southend, Essex, England*

*September 1996 - September 2000 (Part-time)*

Designed and developed a call logging software suite for use in small businesses and hotels from initial conception through to installation including post installation support.

### **Computer Analyst/Programmer**

*HSBC Bank PLC, London, EC3R 6HA, England*

*October 1995 - October 1997*

Working in Payments and Electronic Banking Systems I worked on a number of the payment systems as well as the front end design and the training systems for the new query system. This also included the actual delivery of the end user training to the user department of over 100 people.

## **Education**

### **PhD in Multi-Point Haptics**

*Department of Cybernetics, University of Reading, Reading, RG6 6AY, England  
January 2002 - June 2011 (Part-time)*

See Employment History, Multi-Point Haptics for details of the research undertaken.

### **BSc (Hons) Cybernetics and Control Engineering**

*Department of Cybernetics, University of Reading, Reading, RG6 6AY, England  
September 1997 - June 2000*

Graduated with an Upper Second class (2.1) Honours degree.

Course included:

- Control Engineering including fuzzy logic, adaptive, predictive, state-space, non-linear and optimal control.
- Electronic Engineering including DSP, analogue and digital electronics.
- Computer programming, modelling and simulation.

## **Publications**

### **Games Related**

#### ***Developed whilst at Ubisoft***

- Far Cry 6 (October 2021)
- Watchdogs Legion (October 2020)
- Watchdogs 2 (November 2016)

#### ***Developed whilst at Codemasters***

- F1 2015 (July 2015)
- DiRT Rally (April 2015)
- GRID Autosport (June 2014)
- GRID 2 (May 2013)
- DiRT: Showdown (May 2012)
- DiRT 3 (May 2011)
- F1 2010 (September 2010)
- Colin McRae: DiRT 2 (September 2009)
- Race Driver: GRID (May 2008)
- Colin McRae: DiRT (June 2007)

#### ***Assisted whilst at Codemasters***

- Showdown Arcade (Cabinet) (2015)
- F1 Online (2013) (closed down)
- GRID Arcade (Cabinet) (2010)

#### ***Developed whilst at Asylum Entertainment***

- Curse: The Eye of Isis (October 2003)
- Dinosaur World (October 2002)

### **AI Articles**

1. An Architecture Overview for AI in Racing Games,  
Simon Tomlinson, Nic Melder,  
Game AI Pro: Collected Wisdom of Game AI Professionals, 2012, pp471-480
2. Representing and Driving a Race Track for AI Controlled Vehicles,  
Simon Tomlinson, Nic Melder,

- Game AI Pro: Collected Wisdom of Game AI Professionals, 2012, pp481-490
3. Racing Vehicle Control Systems using PID Controllers,  
Nic Melder, Simon Tomlinson,  
Game AI Pro: Collected Wisdom of Game AI Professionals, 2012, pp491-500
  4. The Heat Vision System for Racing AI: A Novel Way to Determine Optimal Track Positioning,  
Nic Melder,  
Game AI Pro: Collected Wisdom of Game AI Professionals, 2012, pp501-506
  5. A Rubber-Banding System for Gameplay and Race Management,  
Nic Melder,  
Game AI Pro: Collected Wisdom of Game AI Professionals, 2012, pp507-512

### **AI Presentations**

- Game/AI Conference Presentation: The Evolution Of Codemasters' Racing AI: From DiRT To Showdown

## **Academic Related**

### **Academic (Haptic) Papers**

1. Multi-Finger Manipulation Physics for Haptic Rendering,  
N. Melder,  
PhD Thesis, University of Reading, 2011
2. Force Shading and Bump Mapping using the Friction Cone Algorithm,  
N. Melder, W. S. Harwin,  
Proceedings of WorldHaptics, 2005 (Pisa)
3. Perceptual Cues for Orientation in Two Haptic Finger Grasp Task,  
S. McKnight, N. Melder, A. L. Barrow, W. S. Harwin, J. P. Wann,  
Proceedings of WorldHaptics, 2005 (Pisa)
4. Psychophysical Size Discrimination using Multi-fingered Haptic Interfaces,  
S. McKnight, N. Melder, A. L. Barrow, W. S. Harwin, J. P. Wann,  
Proceedings of EuroHaptics, 2004 (Munich)
5. Extending the Friction Cone Algorithm for Arbitrary Polygon Based Haptic Objects,  
N. Melder, W. S. Harwin,  
Proceedings of the 12th International Symposium on Haptic Interfaces for Virtual Environments and Teleoperator Systems, 2004 (Chicago)
6. Translation and Rotation of Multi-Point Contacted Virtual Objects,  
N. Melder, W. S. Harwin,  
Proceedings of EuroHaptics, 2003 (Dublin)
7. Improved Haptic Rendering for Multi-Finger Manipulation using Friction Cone based God-Objects,  
N. Melder, W. S. Harwin,  
Proceedings of EuroHaptics, 2002 (Edinburgh)

### **Academic (Haptic) Presentations**

- Paper Presentation at Eurohaptics, 2002 (Edinburgh, UK)
- Paper Presentation at Eurohaptics, 2003 (Dublin, Ireland)
- Paper Presentation at The International Symposium on Haptic Interfaces for Virtual Environments and Teleoperator Systems, 2004 (Chicago, USA)
- Poster Presentation at WorldHaptics, 2005 (Pisa, Italy)

### **Demonstrations (Haptic)**

- Science Museum, 2005 (London, UK)
- GIST Seminar Series, 2005 (Glasgow, UK)
- ICDVRAT, 2004 (Oxford, UK)
- Corso Euro/Verano, 2004 (San Sebastian, Spain)

- Venture Fest, 2004 (Oxford, UK),
- EuroHaptics, 2004 (Munich, Germany)
- Haptic Symposium, 2004 (Chicago, USA)

### ***Lecturing***

- Haptic Rendering (2004/2005)  
2 Lectures as part of the 4th year course 'Haptics and Robotics'. Spring term.
- Haptic Interfaces for Virtual Reality and Teleoperation (2004/2005)  
Introductory lecture as part of the MSc Course. Autumn Term.

### **Testimonials**

"I worked with Nic on DiRT Showdown, GRID 2, GRID Autosport. It was a pleasure to work alongside Nic and he was always a man who had time to share his vast technical expertise and range of experience with a fresh faced devQA team. Affable but never afraid to speak his mind he was always a great character to have in the studio, suffice to say that without Dr Nic's early guidance our test engineer team would never have been as technically effective during GRID2 or GRID Autosport.

I would gladly work with Nic again in any capacity!"

- Phil Rai, Former Associate Producer at Codemasters, June 2015

### **Referees**

Referees are available upon request.