Real Time, GPU-Accelerated Radar Modeling for Autonomous Vehicle Design

Jeffrey Decker (jeffrey.decker@ansys.com)

Lead R&D Engineer

Ansys

Outline

- Radar simulation uses in Advanced Driver Assistance Systems (ADAS)
- Need for real time radar simulation
  - Ray tracing for radar applications
  - Understanding range-Doppler images
- GPU acceleration to achieve real time
  - Ray tracing, Materials, Image Generation, Antenna Models, Velocity, Multi-Bounce
- Future work

- Collaborators
  - Bob Kipp (Project Lead)
  - Jeffrey Decker (Technical Lead)
  - Stefano Canta
  - Daniel Rey

Role of Simulation in Autonomous Vehicle Design

- **Sensor Design**
  - Sensor Components and System
  - Sensor Response to Environment
- **System Design and Testing**
  - Sensing Environment and Actors
  - Perception
  - Decision Making
  - Actuation
  - Vehicle Dynamics
  - Update Actors
- M/Billion-mile Regression Testing
  - Changes to fielded systems
  - Requires real-time+ simulation

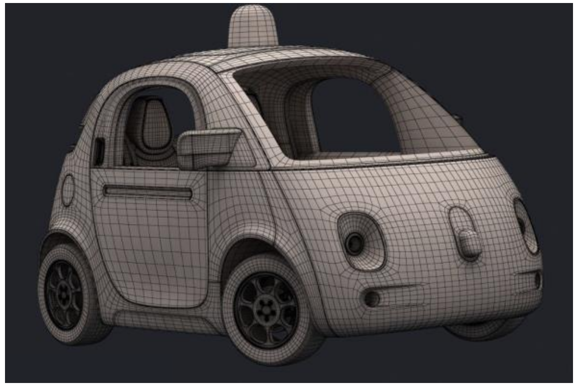
Driven by Feb 2018

20 Million Road Miles

Driven by January 2020

2.5 Billion Simulated Miles

Driven in ONE Year



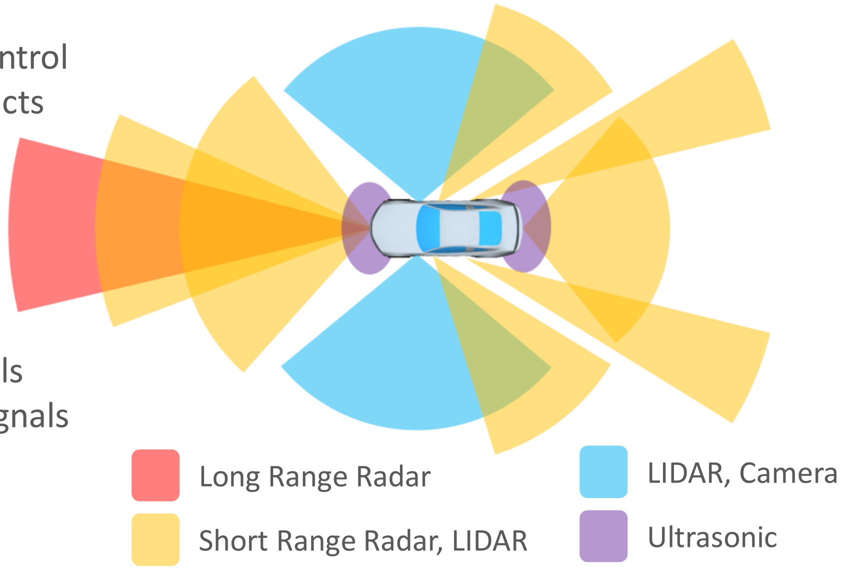
10 Billion Total Simulated Miles

Driven by July 2019

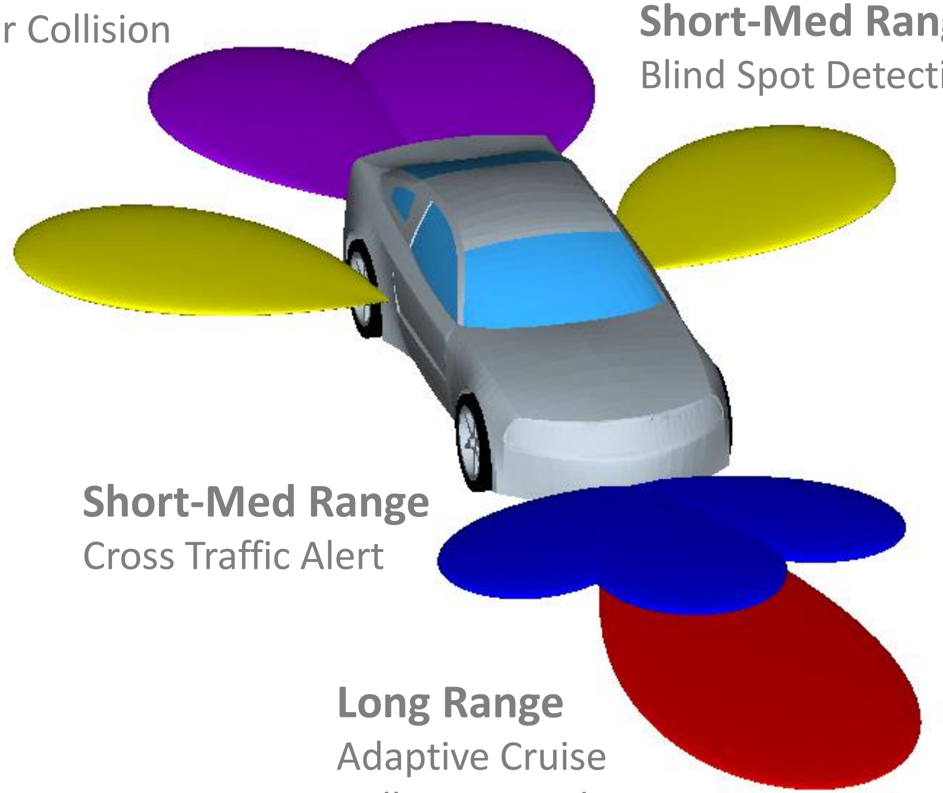
Reference:
https://medium.com/waymo/waymo-reaches-5-million-self-driven-miles-61fba590fafe
https://techcrunch.com/2019/07/10/waymo-has-now-driven-10-billion-autonomous-miles-in-simulation/
https://fortune.com/2020/01/07/googles-waymo-reaches-20-million-miles-of-autonomous-driving/

Sensor Suite for Autonomous Vehicles

Sensor Tasks
Adaptive cruise control
Long distance objects
Lane detection
Blind spot
Lane changing
Rear collision
Side view
Pedestrians/animals
Traffic signs and signals
and more …



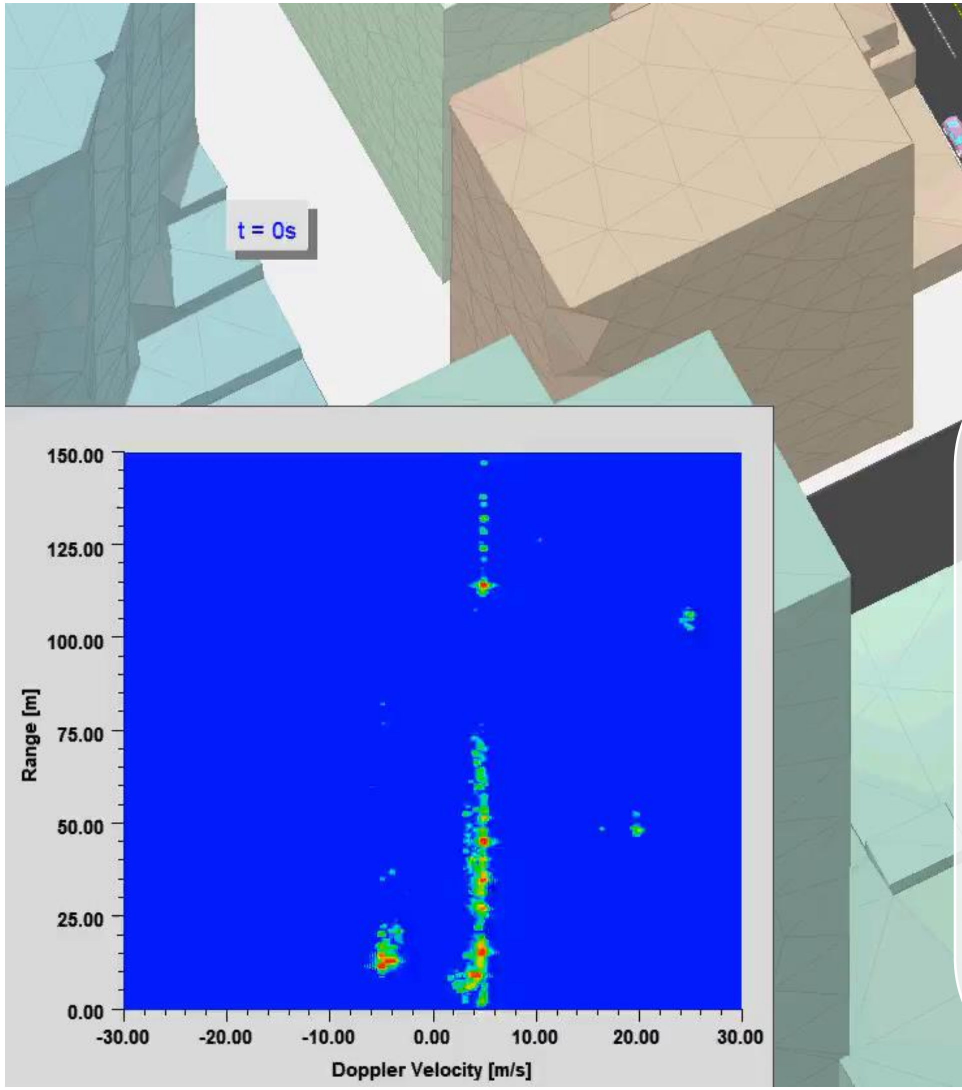
Typical Automotive Radar Sensors



Performance Aspect	Human	Autonomous Vehicle		
		Radar	Lidar	Camera
Object detection	Good	Good	Good	Fair
Object classification	Good	Poor	Fair	Good
Distance estimation	Fair	Good	Good	Fair
Edge detection	Good	Poor	Good	Good
Lane tracking	Good	Poor	Poor	Good
Visibility range	Good	Good	Fair	Fair
Poor weather performance	Fair	Good	Fair	Poor
Dark or low illumination performance	Poor	Good	Good	Fair

Schoettle, Brandon, "Sensor Fusion: A Comparison of Sensing Capabilities of Human Drivers and Highly Automated Vehicles", Sustainable Worldwide Transportation, U of Michigan, 2017

CPU-Based Radar Sensor Solution: Ansys HFSS-SBR+ 2019 R2



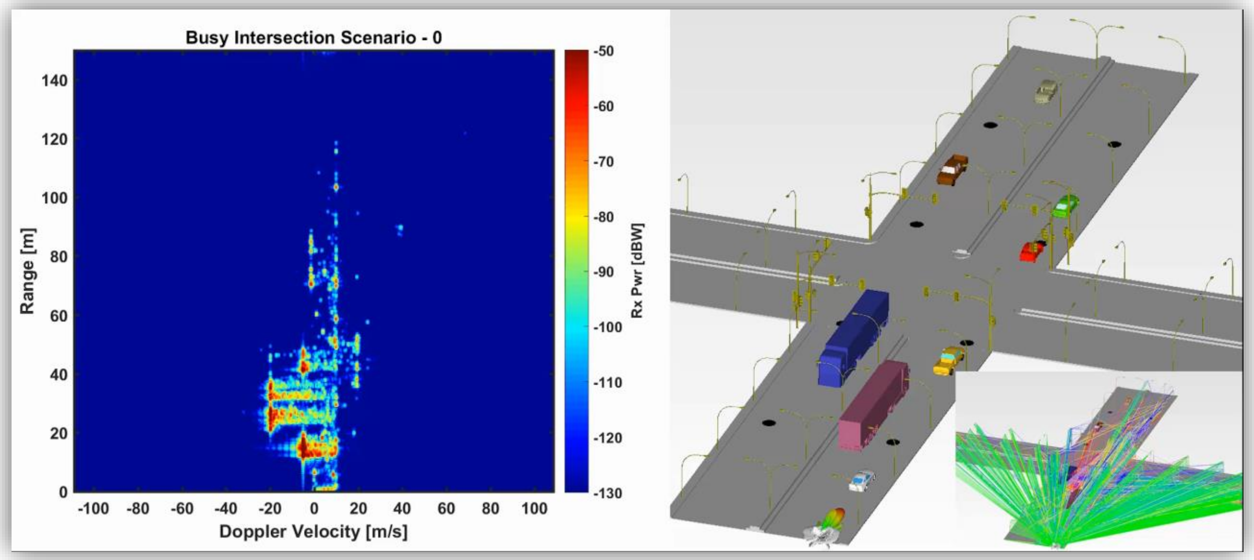
This 12 sec scenario requires
<u>14 HOURS</u> to simulate on a workstation!

What's it take to simulate in REAL TIME?

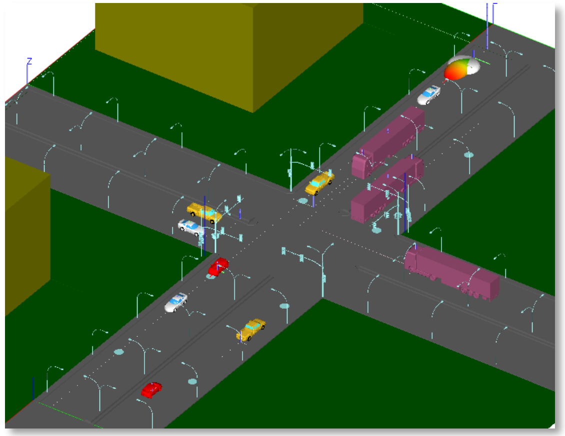
30 frames per second!

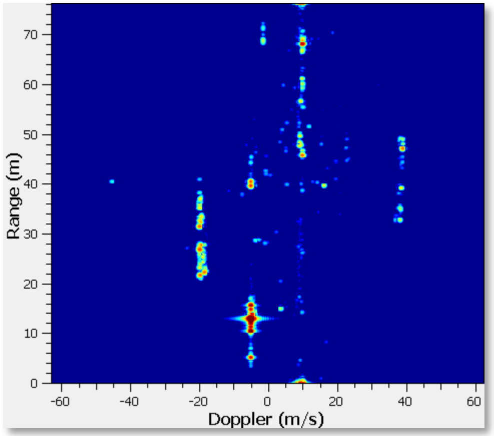
Background on Shooting & Bouncing Rays (SBR+) Solution

- SBR+:
  - Shooting-and-Bouncing Rays (SBR) method
  - PLUS (+) unique Ansys enhancements
- SBR+ in the radar sensor application
  - Launch millions of rays from transmitter (Tx)
  - Rays bounce 1+ more times
  - “Paint” EM currents at each bounce
  - Radiate currents to receiver (Rx) antenna(s)
- Key point: ray bounce contributions are tracked as waves
  - Constructively or destructively interfere depending on path length and frequency
  - → Realistic Physics



Algorithmic and Hardware Acceleration Breakthrough





1 Frame
Center Frequency = 76.5 GHz
Bandwidth = 502 MHz
256 frequency samples
4 ms processing interval, 256 chirps

General-Purpose SBR+ > 10 hours → **Algorithmic Improvements < 1 min** → **GPU Real Time Solver ~12 ms** NVIDIA Quadro GV100

Algorithmic Acceleration = over 600 X

Hardware Acceleration ≈ 5000 X

Composite Acceleration = 3+ Million X !!

Ansys

Goal: Real Time GPU Radar Imaging

- Starting Point for GPU Solver:
  - CPU: 30+ sec/image on 24-cores @ 3 GHz (2019 workstation)
  - Goal: 30 frames/sec → 0.0333 sec/image
  - Need: 910x speedup
- Status:
  - 100% GPU accelerated solver
  - Generates images at faster than real time
  - Multi-bounce, multi-Rx, materials
  - 1500-3000x speedup
- Ongoing Work:
  - More Validation
  - Finish transmitted rays
  - More optimization

	Receivers	CPU	GPU	Speedup
Radar Image Generation	1 Channel	30+ sec	10-20 ms	1500-3000 x
	8 Channels	240+ sec	40-45 ms	5300-6000 x

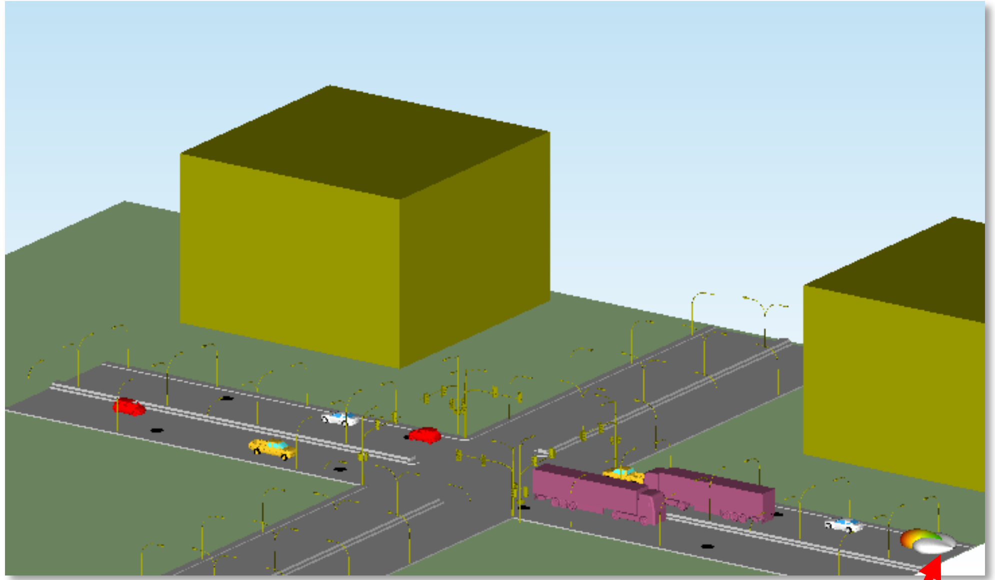
Busy Intersection Test Scene

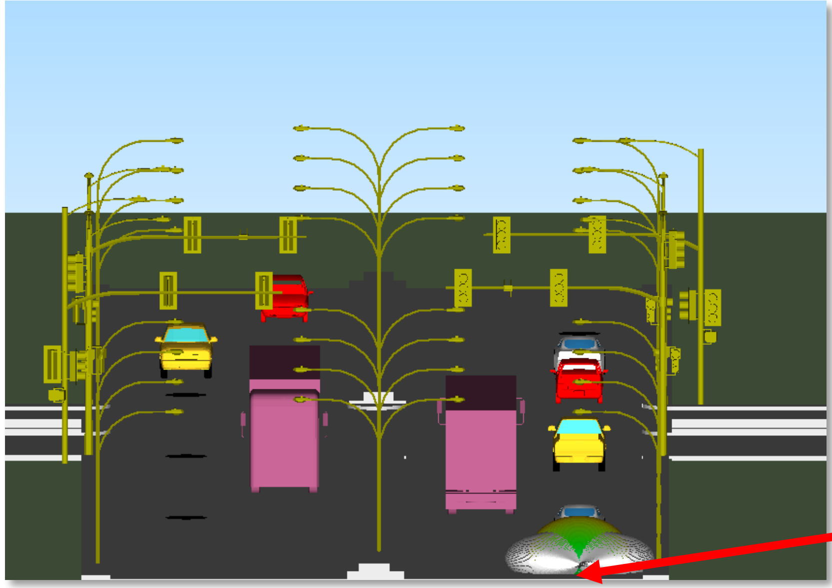
120m x 150m, 9 vehicles, 48 lights, 6 stop lights, curbs, buildings, medians

Level of Detail: 547,000 facets

Radar – mounted on the "EGO" vehicle

- 76.4 – 76.6 GHz, 267 frequency samples
- 4 ms CPI, 256 pulses
- Range: Res = 0.75 m, 0 – 200 m
- Velocity: Res = 0.49 m/s, ± 49 m/s
- Range-Doppler Image
    - 1024 x 1024 pixels



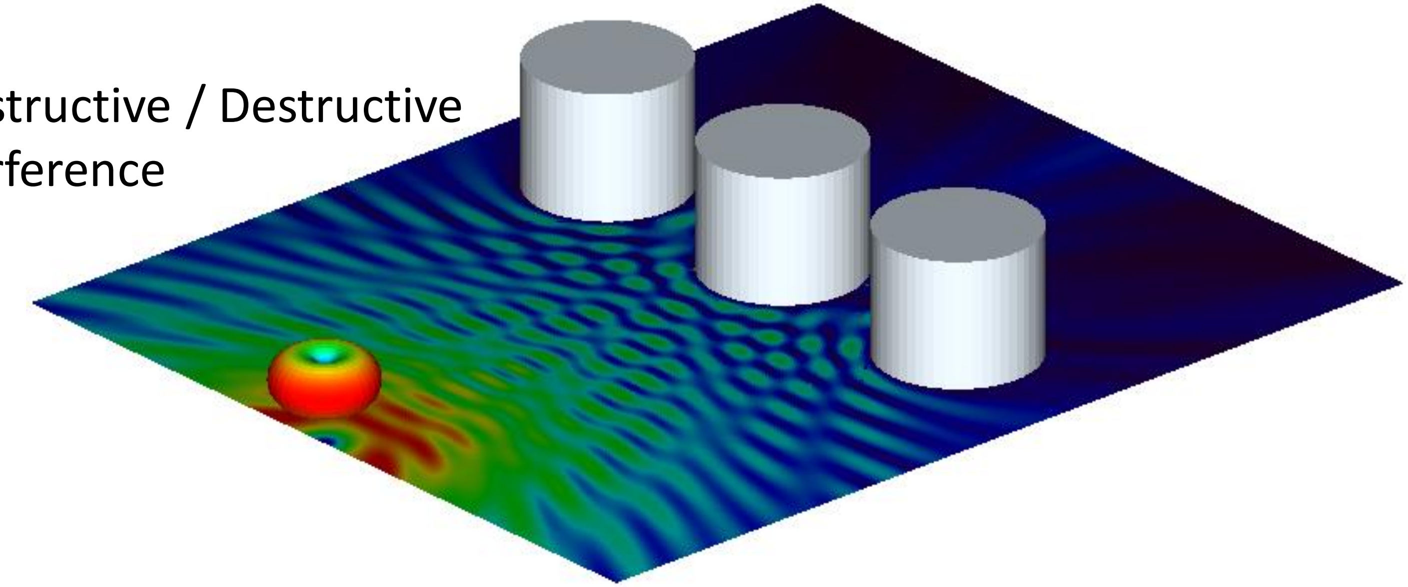


Ray Tracing for Radar versus Optical Simulation

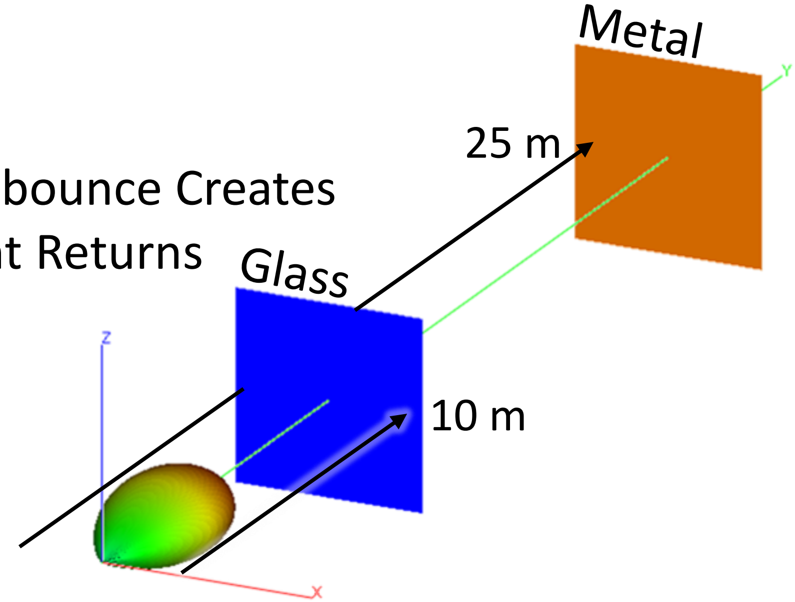
Radar requires different quantities:
- phase, E&H fields, currents, etc.

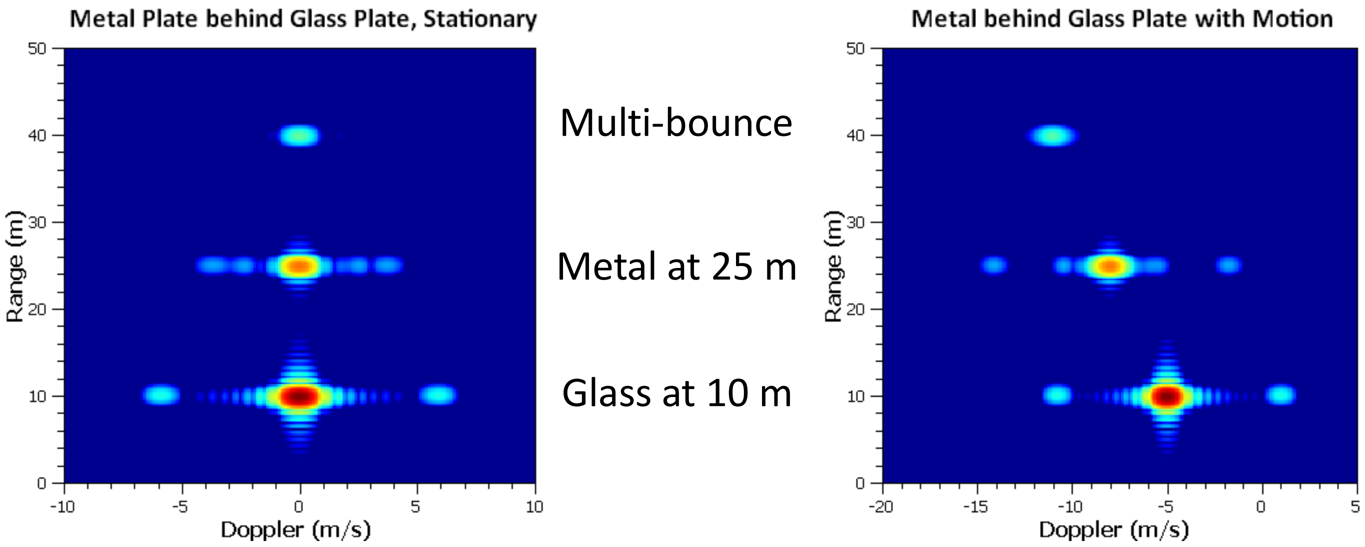
Different effects:
- interference patterns
- multi-bounce returns
- velocity affects frequency (Doppler)

Constructive / Destructive Interference

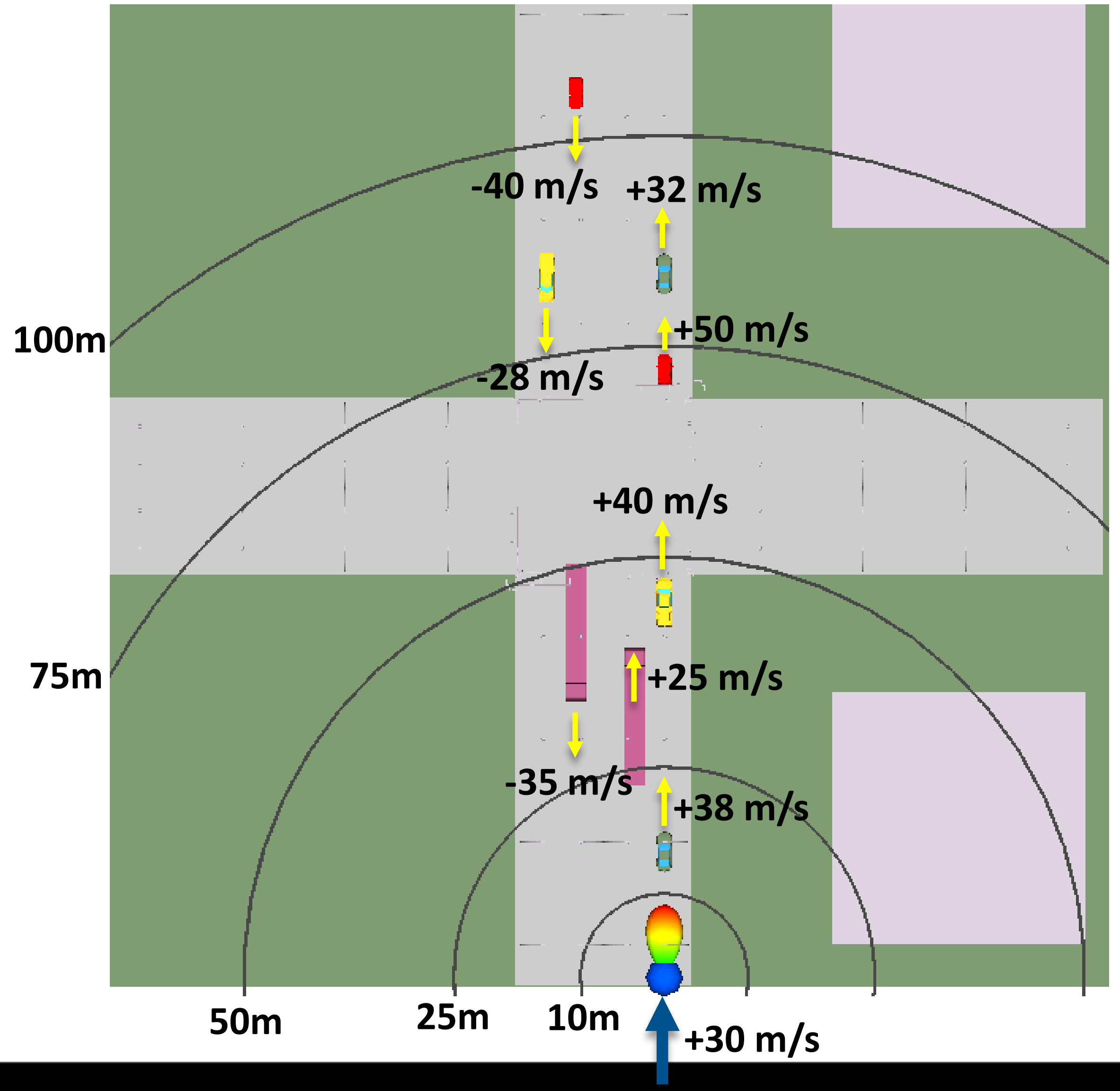


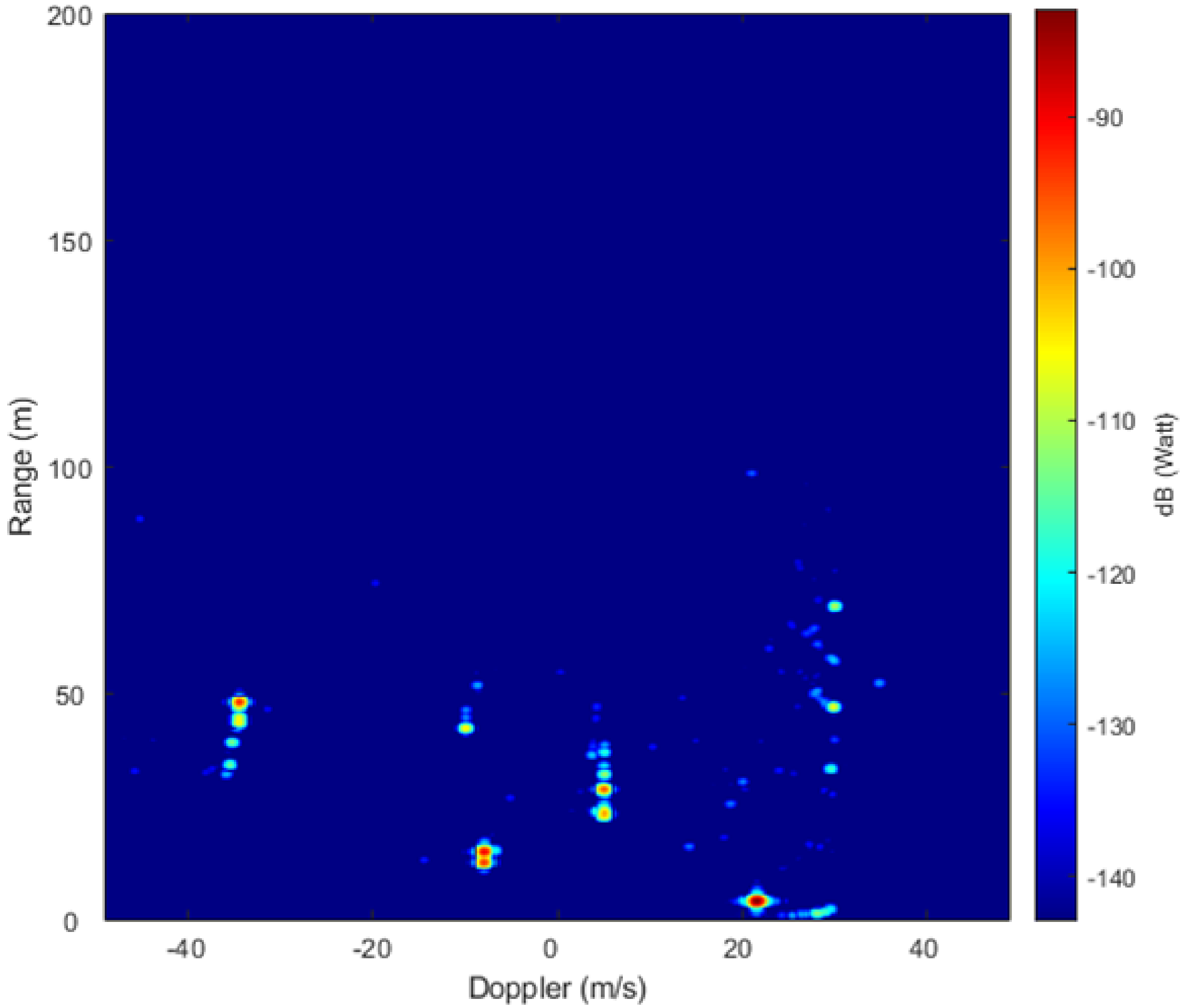
Multi-bounce Creates Distant Returns



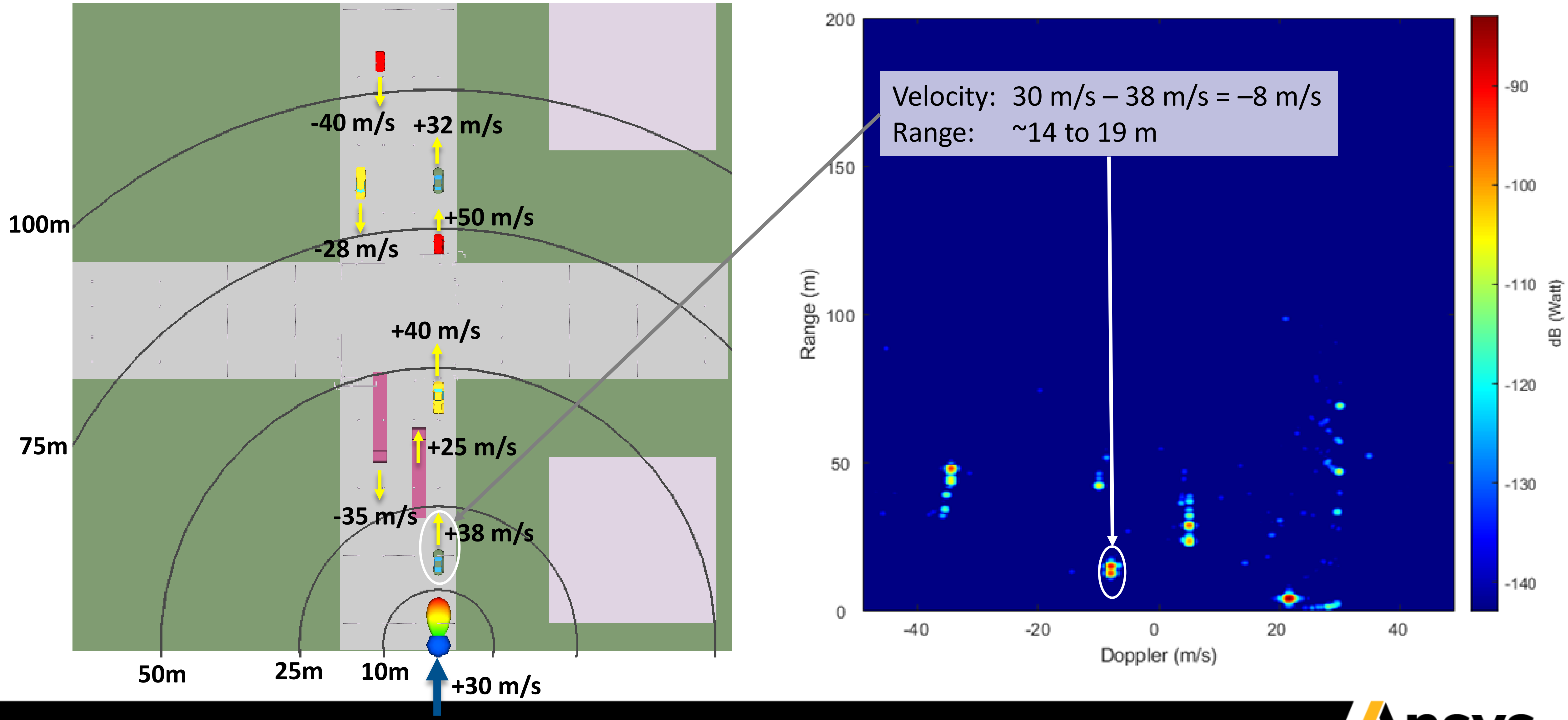


Interpreting Range-Doppler Imagery

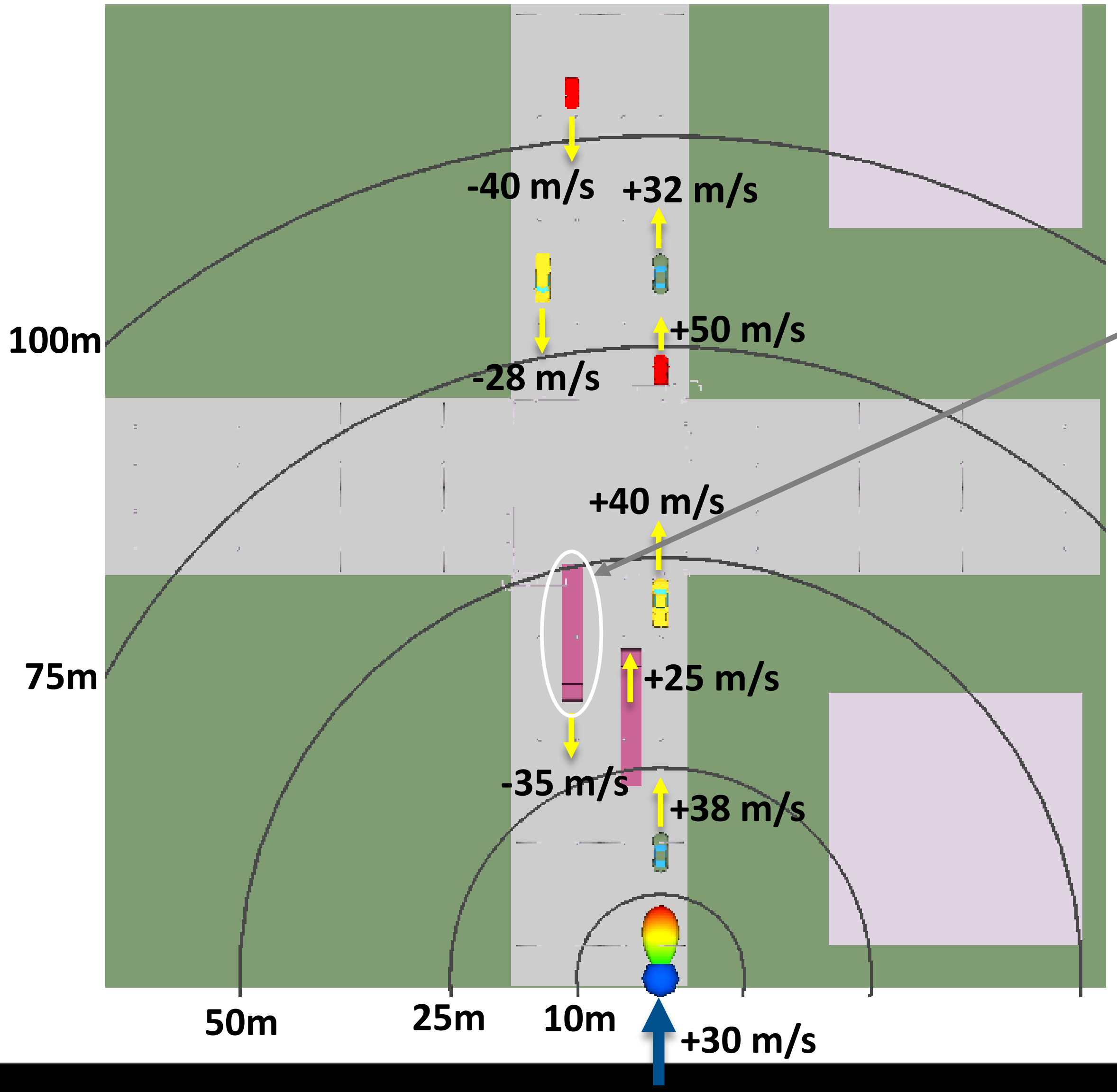


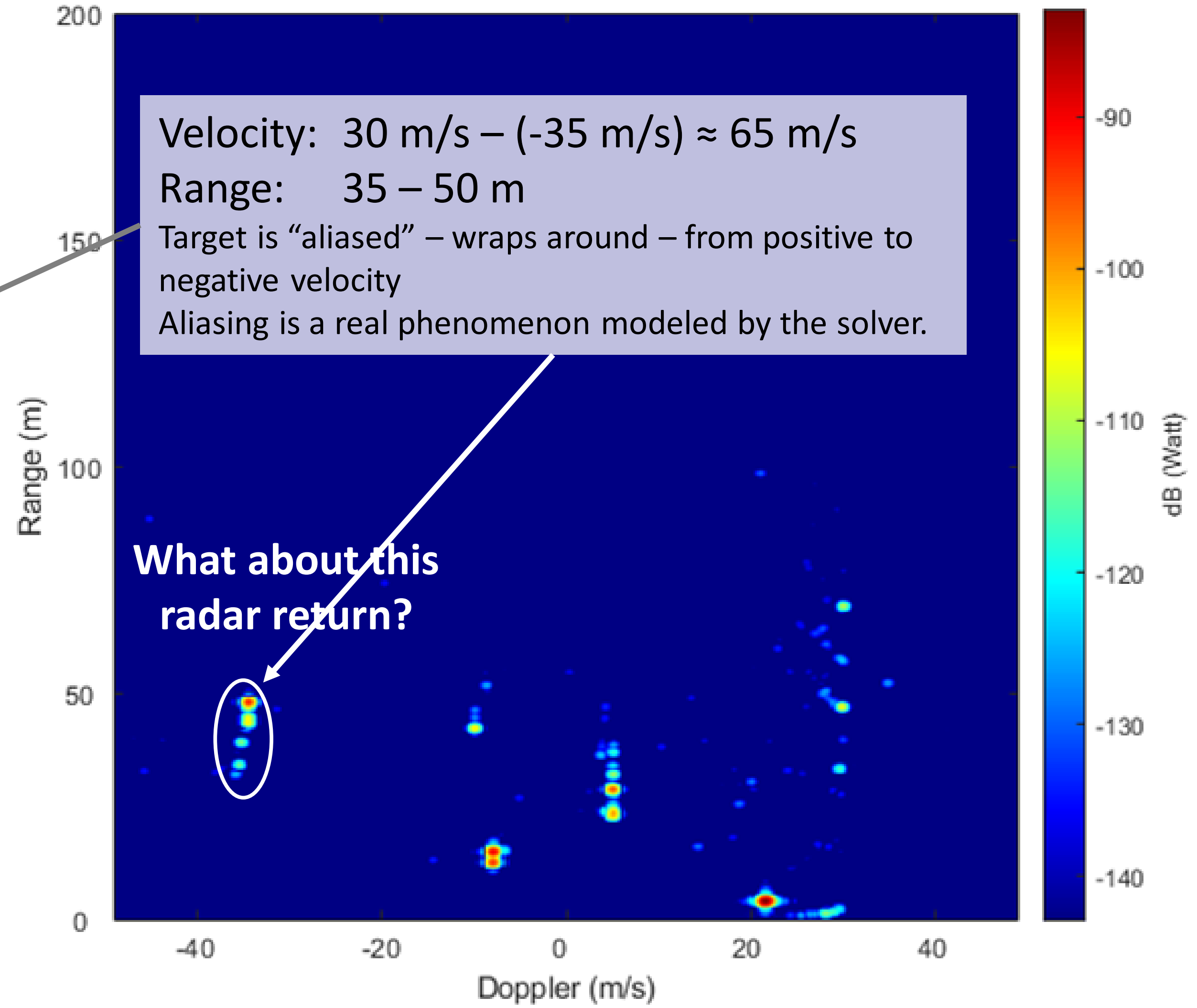


Interpreting Range-Doppler Imagery



Interpreting Range-Doppler Imagery





Solver Computational Flow and GPU Acceleration

- Acceleration
  - 100% GPU accelerated
  - Almost no device-host transfers
  - Use NVIDIA provided libraries for anything that fits
  - Redesign existing GPU kernels for automotive radar
- Hardware
  - Use latest GPUs
    - Quadro GV100, RTX 6000/8000
    - Large memory processes more data without I/O or extra loops

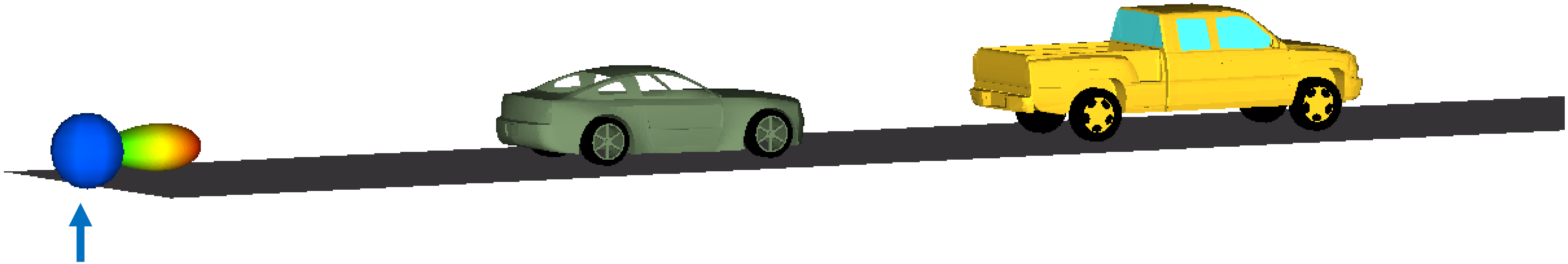
	OptiX	CUDA	cuFFT
Solver Initialization			
Time Loop			
Update Positions, Orientations, & Velocities	×	×	
Multi-Bounce Loop			
Shoot Rays	×	×	
Process Hits		×	
Compute Fields & Currents		×	
Radiate to Rx Antennas		×	
Form Range-Doppler Image		×	×

Thank you to NVIDIA for providing high-end GPUs for development!!!

Ansys

Ray Shooting for Radar Modeling – Reflected, Transmitted, Multibounce Rays

Scenario: Ego Vehicle following two cars

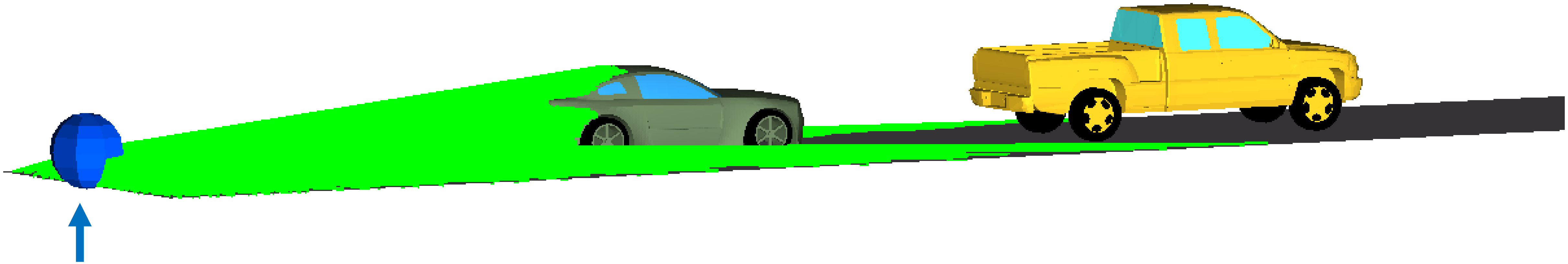


Ego Vehicle Radar

Ansys

Ray Shooting for Radar Modeling – Reflected, Transmitted, Multibounce Rays

Scenario: Ego Vehicle following two cars

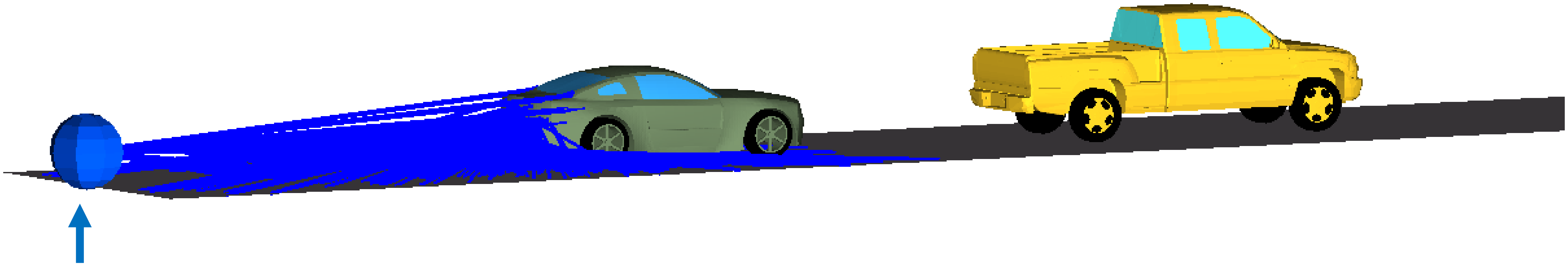


Ego Vehicle Radar

Rays: 1 reflection, only line-of-sight visible

Ray Shooting for Radar Modeling – Reflected, Transmitted, Multibounce Rays

Scenario: Ego Vehicle following two cars

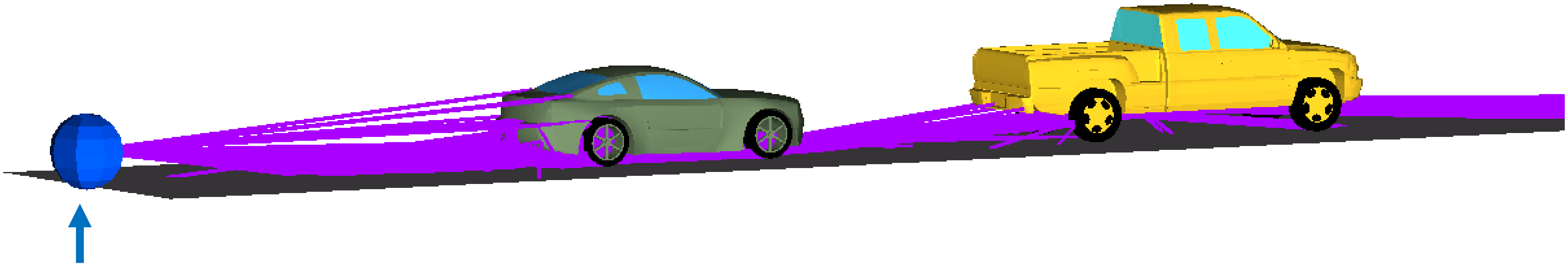


Ego Vehicle Radar

Rays: 2 reflections, ground bounce effects

Ray Shooting for Radar Modeling – Reflected, Transmitted, Multibounce Rays

Scenario: Ego Vehicle following two cars



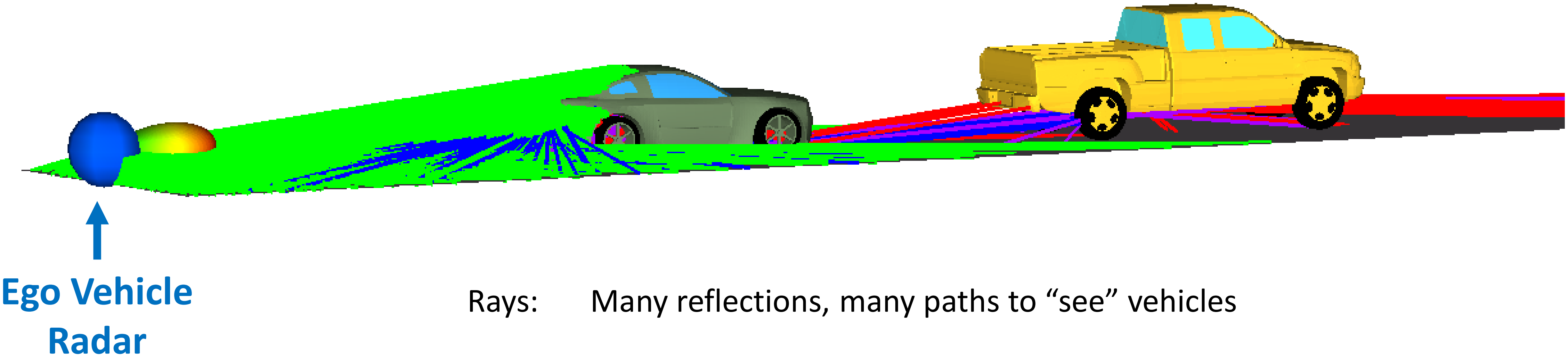
Ego Vehicle Radar

Rays: 3 reflections, truck two vehicles in front is visible

Ansys

Ray Shooting for Radar Modeling – Reflected, Transmitted, Multibounce Rays

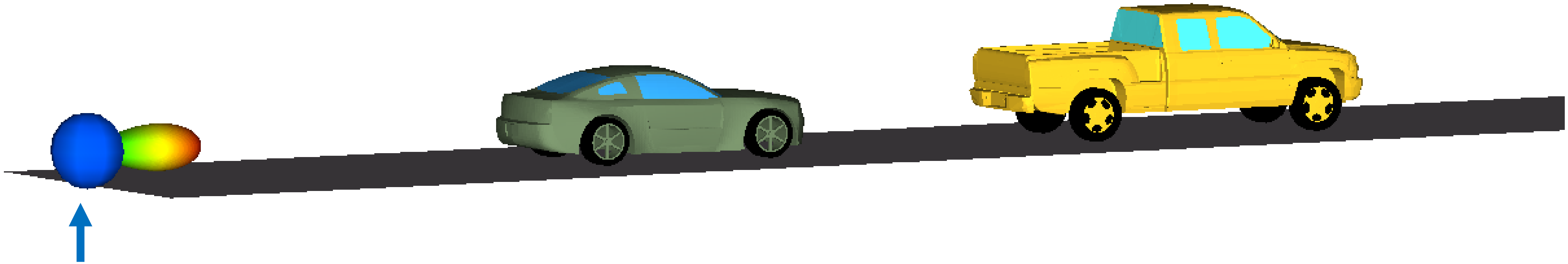
Scenario: Ego Vehicle following two cars



Rays: Many reflections, many paths to "see" vehicles

Ray Shooting for Radar Modeling – Reflected, Transmitted, Multibounce Rays

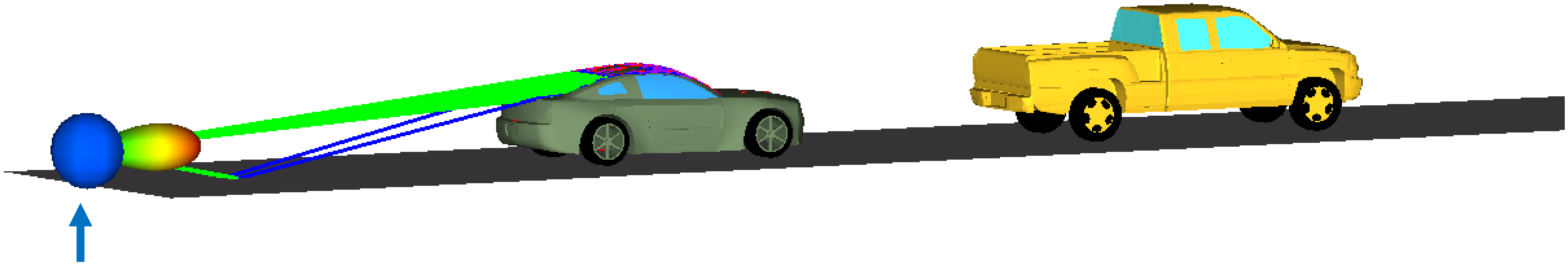
Scenario: Ego Vehicle following two cars



Ego Vehicle Radar

Ray Shooting for Radar Modeling – Reflected, Transmitted, Multibounce Rays

Scenario: Ego Vehicle following two cars



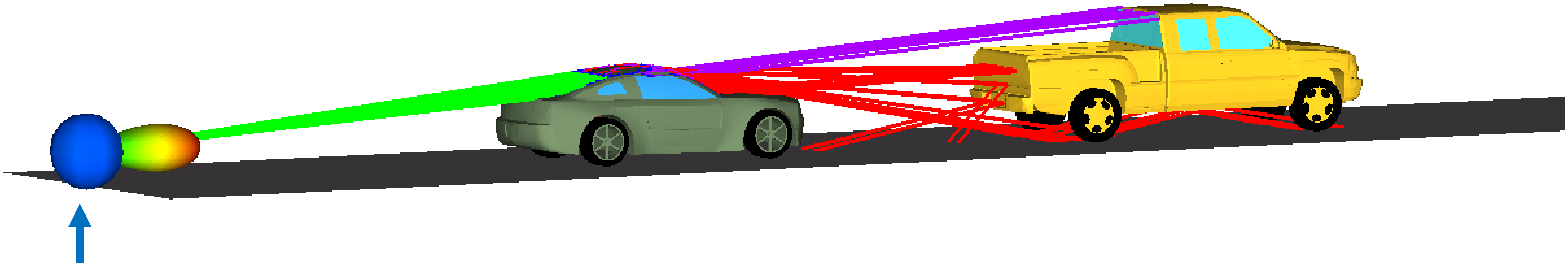
Ego Vehicle Radar

Rays: 1 transmission, see through rear windshield

Ansys

Ray Shooting for Radar Modeling – Reflected, Transmitted, Multibounce Rays

Scenario: Ego Vehicle following two cars

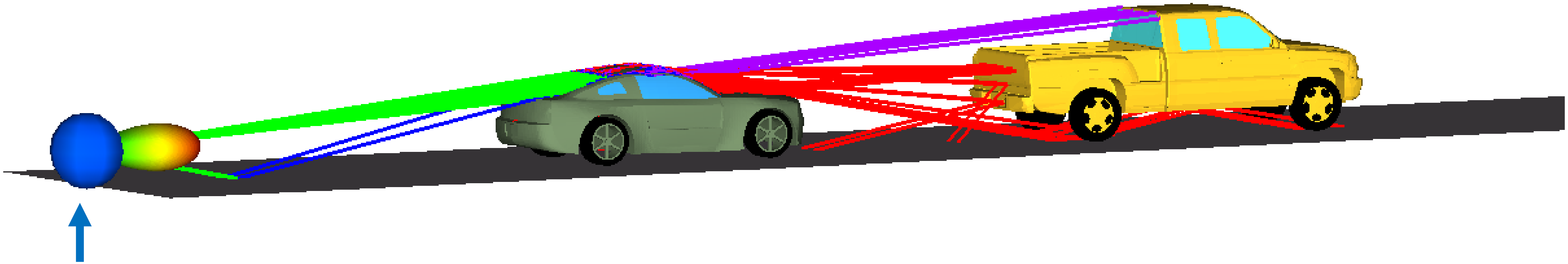


Ego Vehicle Radar

Rays: 2 transmissions, truck is visible

Ray Shooting for Radar Modeling – Reflected, Transmitted, Multibounce Rays

Scenario: Ego Vehicle following two cars



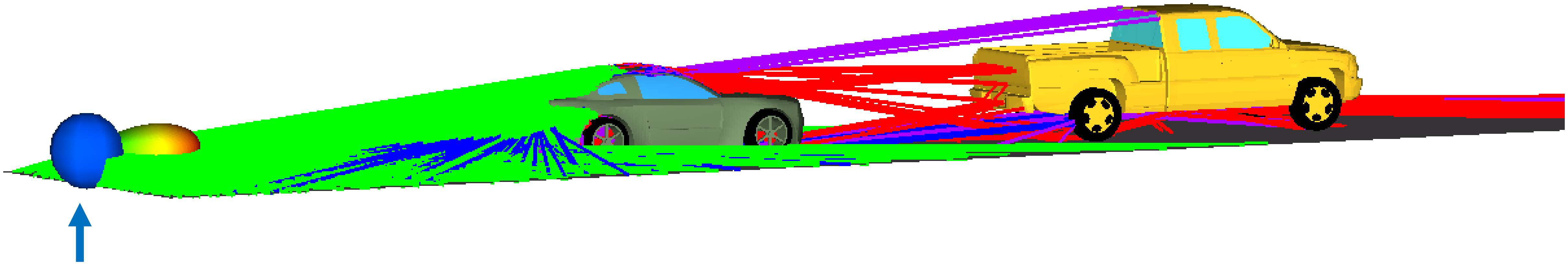
Ego Vehicle Radar

Rays: Many transmitted paths

Ansys

Ray Shooting for Radar Modeling – Reflected, Transmitted, Multibounce Rays

Scenario: Ego Vehicle following two cars



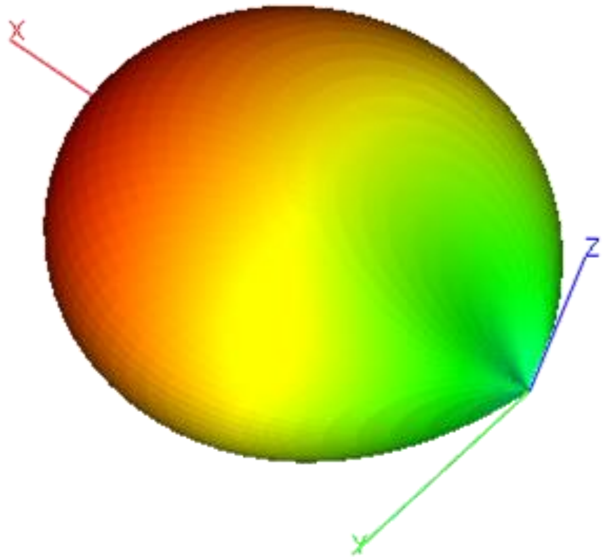
Ego Vehicle Radar

Rays: Reflected and transmitted rays model propagation of radar energy
Multiple paths are critical for physically accurate simulation

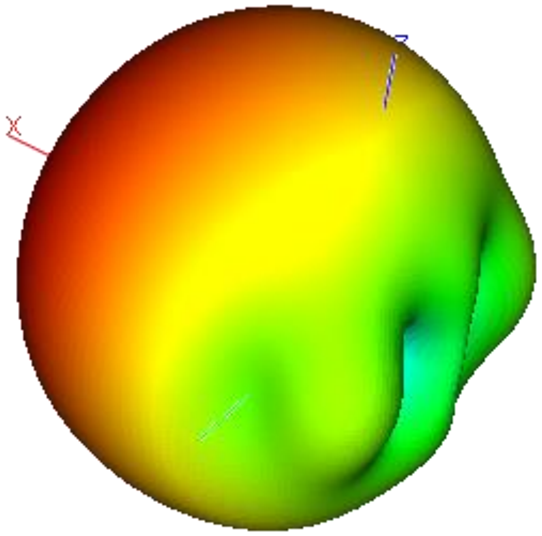
Radar Antenna Modeling

- Solver supports two antenna models
- Parametric antenna beam
  - Formula-based
  - Easy to set up: two beam widths
  - No data file required
- Far field antenna pattern
  - File-based
  - Real antenna patterns
  - Includes side lobes
  - Model the effect of antenna mounted on vehicle

Parametric Beam

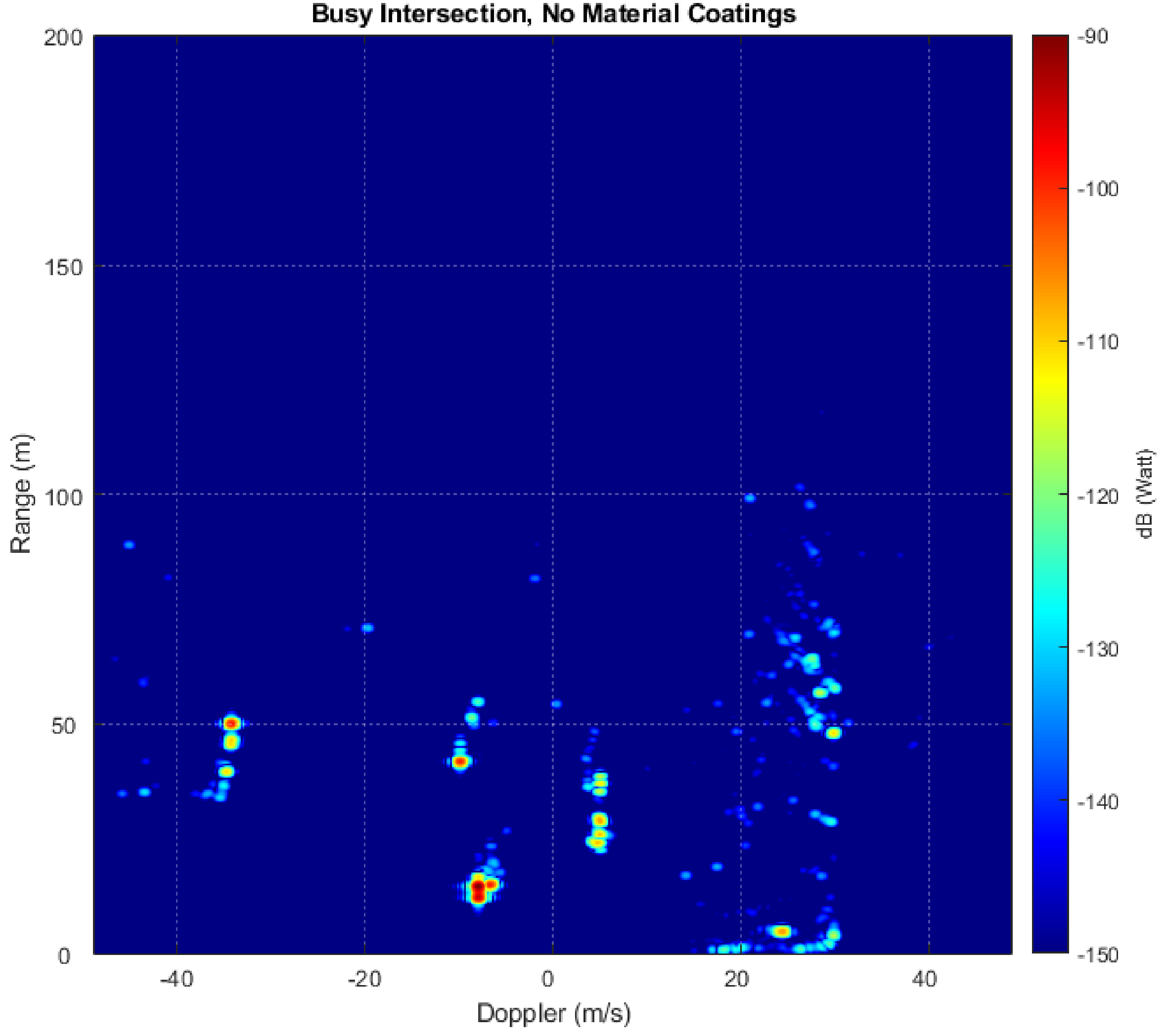


Antenna Pattern



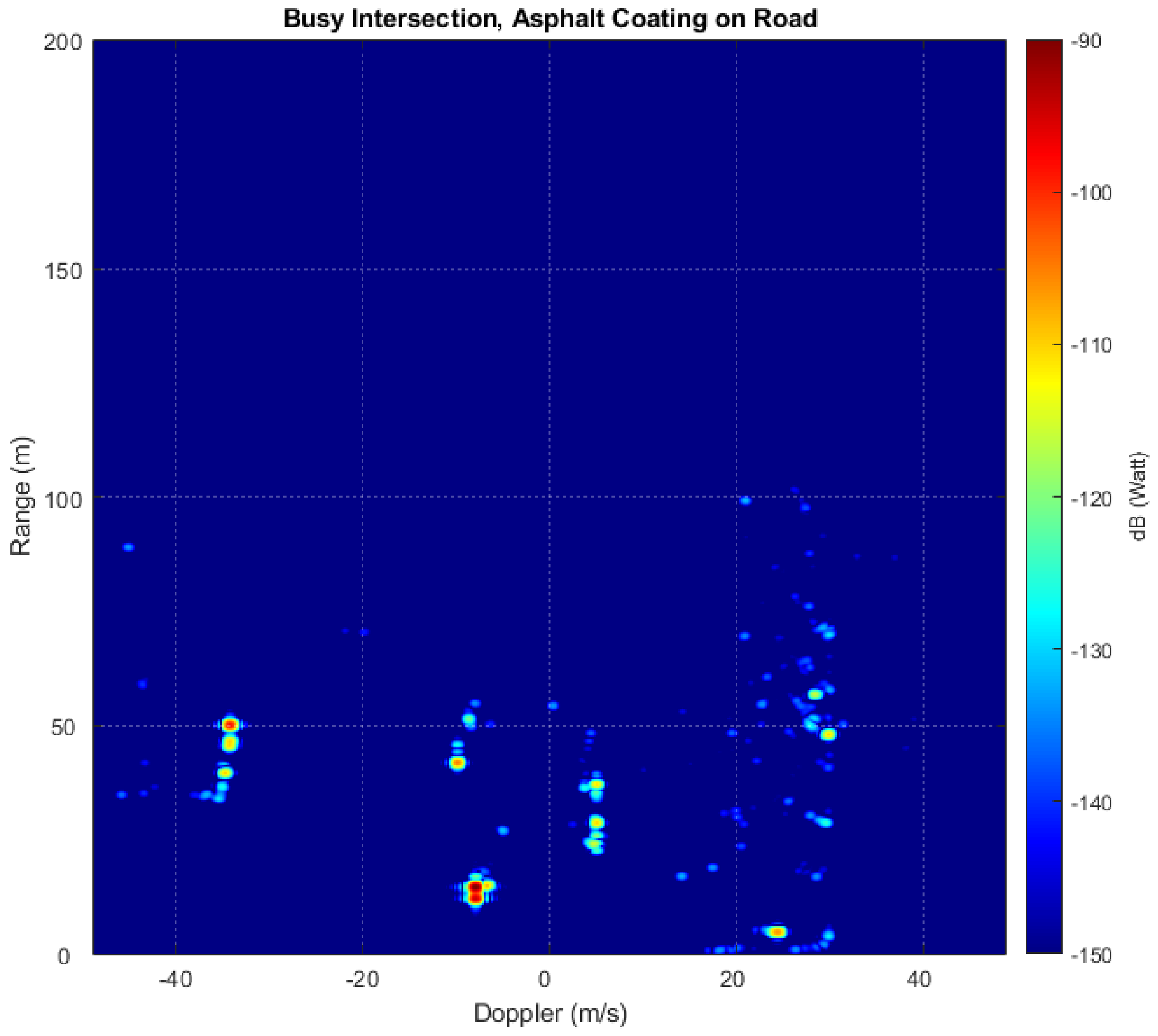
Material Modeling

- Solver supports many material models
  - Metal
  - Layered dielectric materials
  - File tables of reflected-transmitted coefficients
  - Impedance Boundary Condition (IBC)
  - Transmissive materials (e.g. windshield class)
  - Absorber
- Impact on images
  - Peaks are slightly lower with lossy materials
  - Clutter and ground are attenuated
  - Attenuated ground bounce near vehicles
- Rough surface material model will be future addition



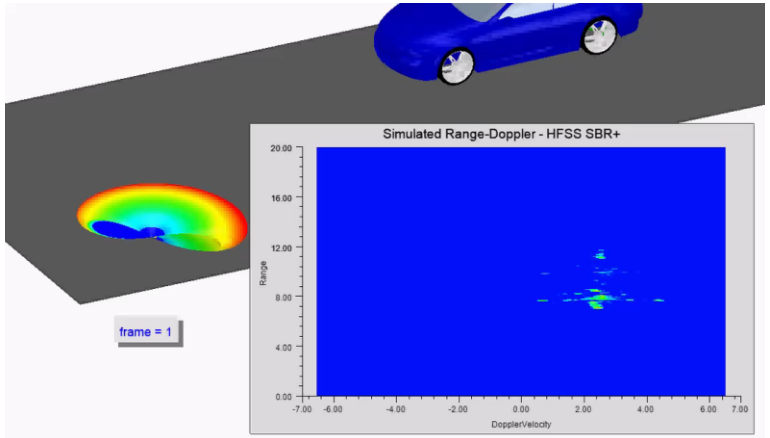
Material Modeling

- Solver supports many material models
  - Metal
  - Layered dielectric materials
  - File tables of reflected-transmitted coefficients
  - Impedance Boundary Condition (IBC)
  - Transmissive materials (e.g. windshield class)
  - Absorber
- Impact on images
  - Peaks are slightly lower with lossy materials
  - Clutter and ground are attenuated
  - Attenuated ground bounce near vehicles
- Rough surface material model will be future addition

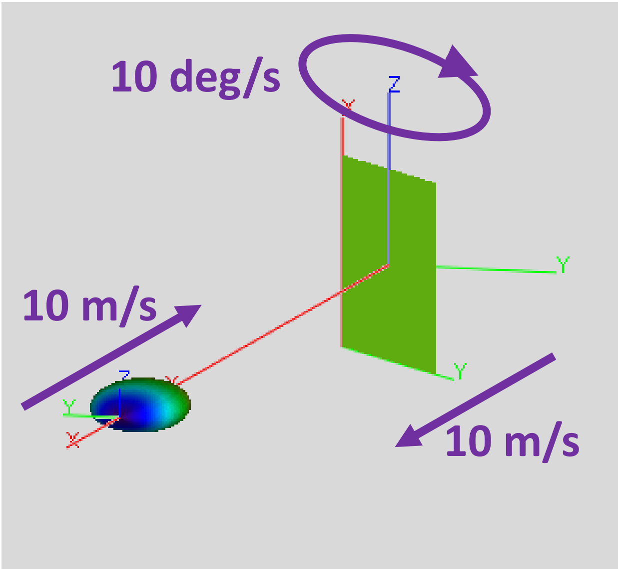


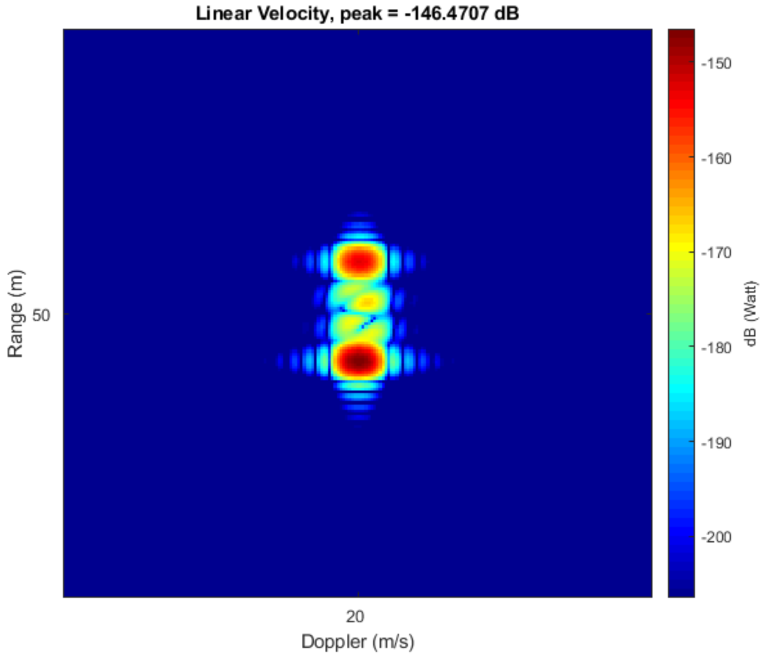
Angular Velocity

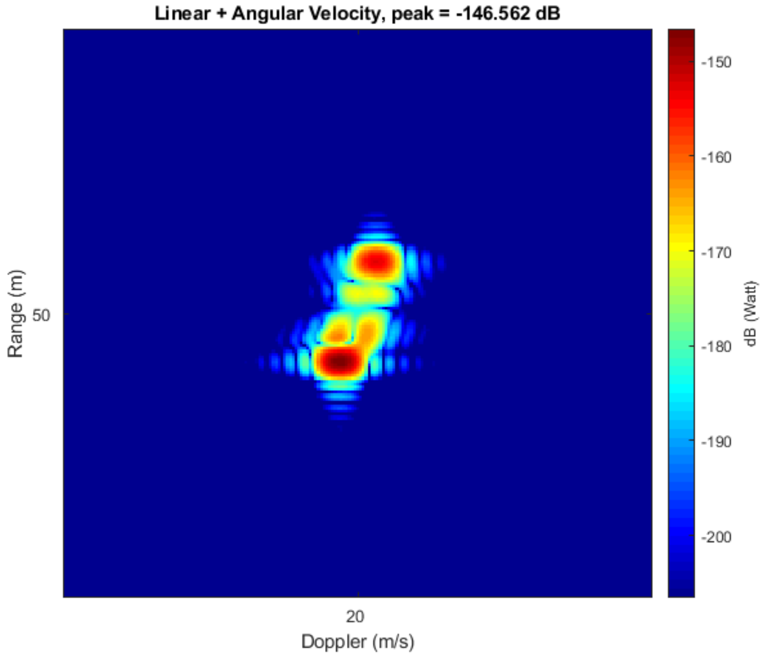
- Solver properly models angular velocity
  - All objects and radars have linear and angular velocity
- Wheel rotation and vehicles turning create Doppler velocity shifts in images



Simulated 77 GHz micro-Doppler radar signatures, Ansys, Shawn Carpenter
Demonstrates Doppler of angular velocity. Not simulated with the GPU solver.

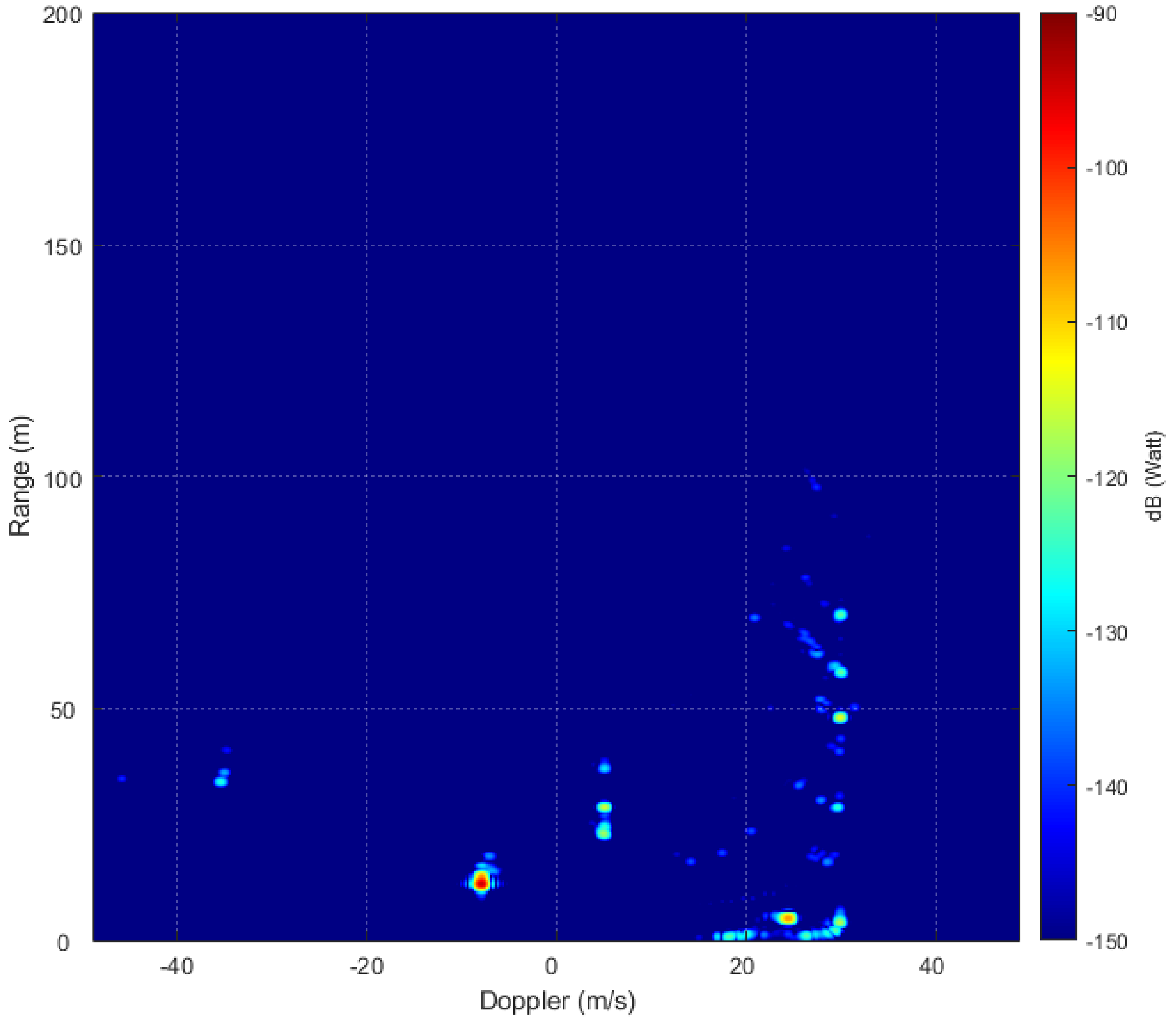






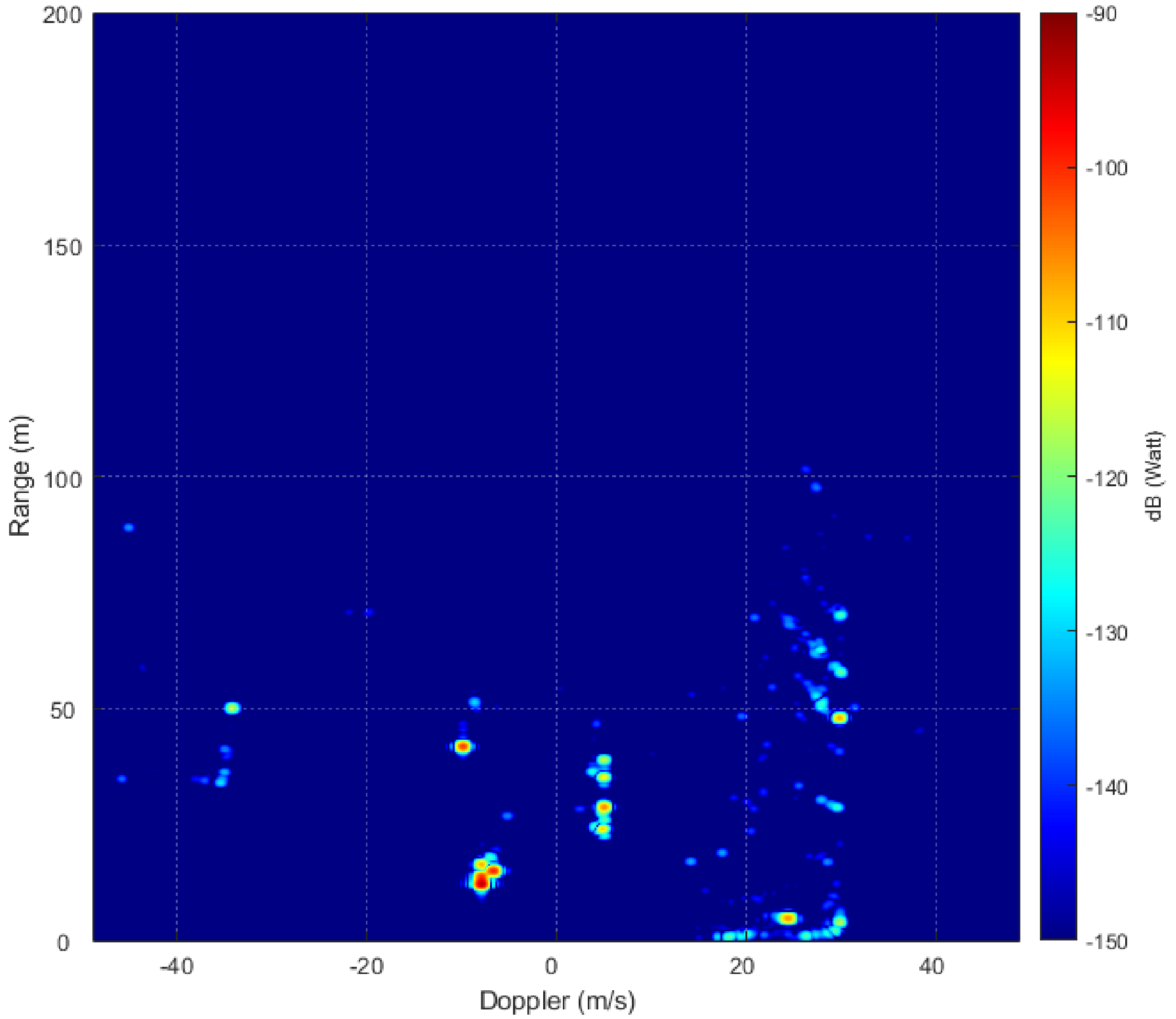
Multi-Bounce

- GPU Solver supports multi-bounce
- Models complex scene interactions
  - Scene-object
  - Object-object
  - Object-scene-object
- Captures physics of radar's beyond line-of-sight sensing



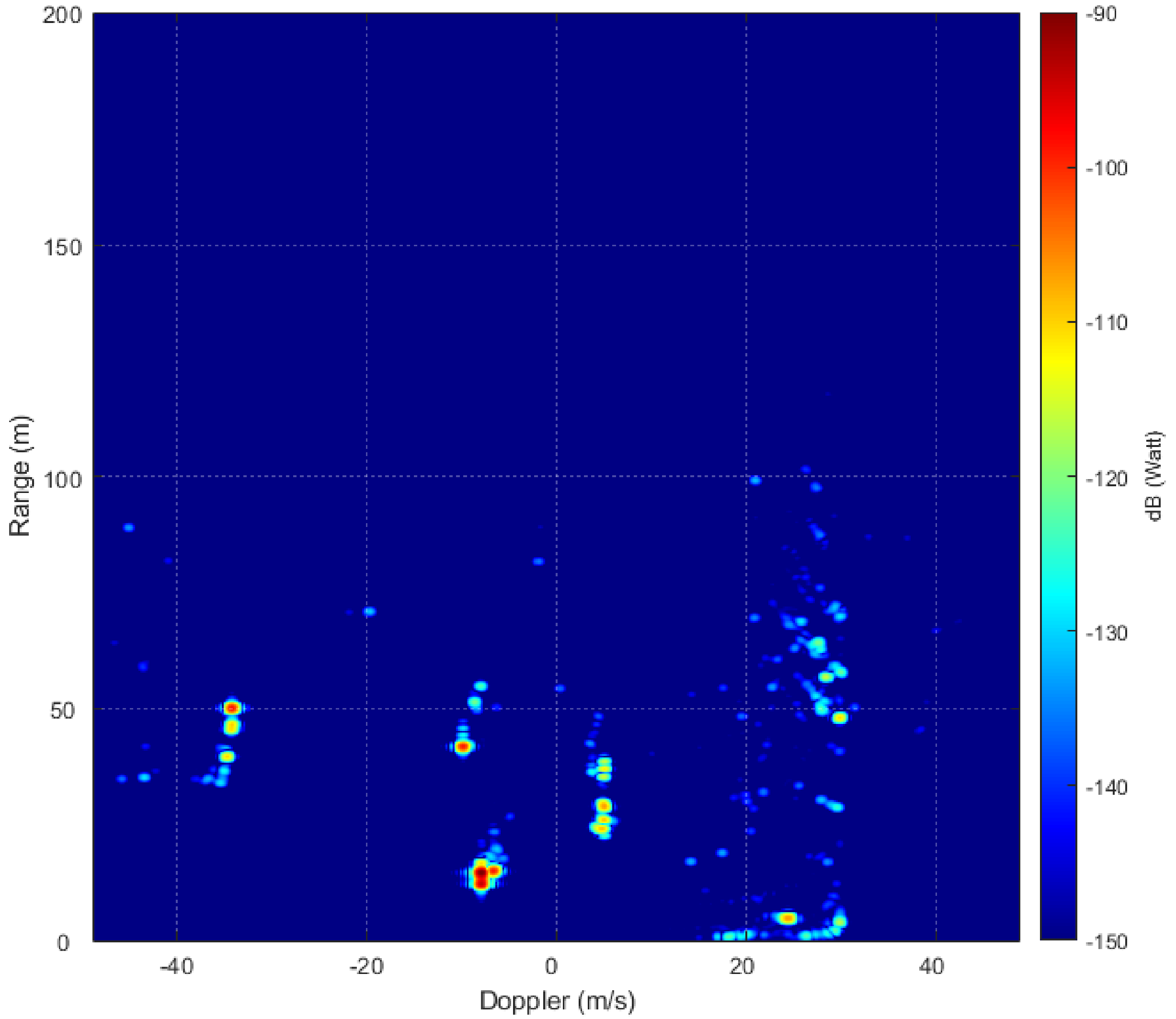
Multi-Bounce

- GPU Solver supports multi-bounce
- Models complex scene interactions
  - Scene-object
  - Object-object
  - Object-scene-object
- Captures physics of radar's beyond line-of-sight sensing



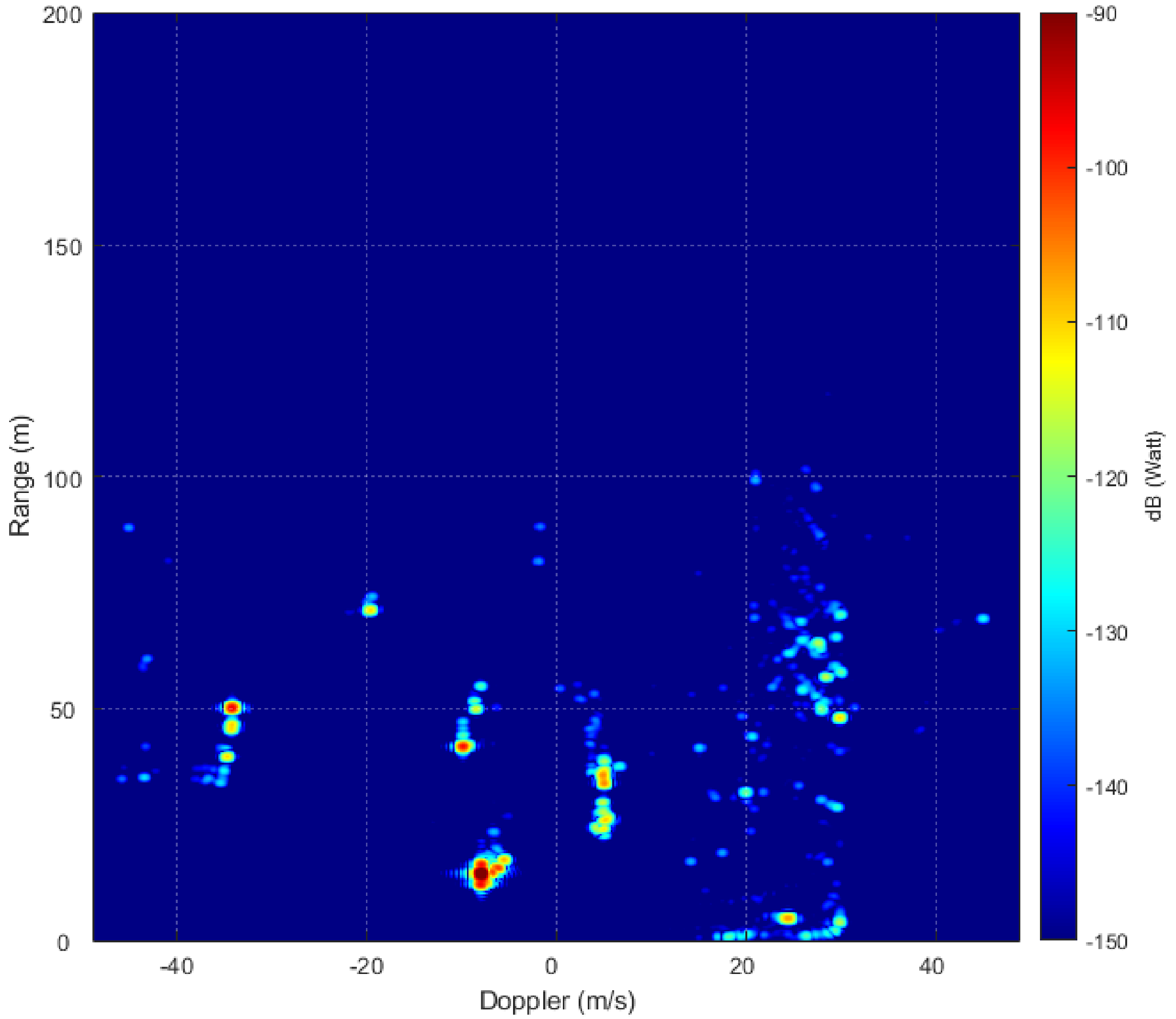
Multi-Bounce

- GPU Solver supports multi-bounce
- Models complex scene interactions
  - Scene-object
  - Object-object
  - Object-scene-object
- Captures physics of radar's beyond line-of-sight sensing



Multi-Bounce

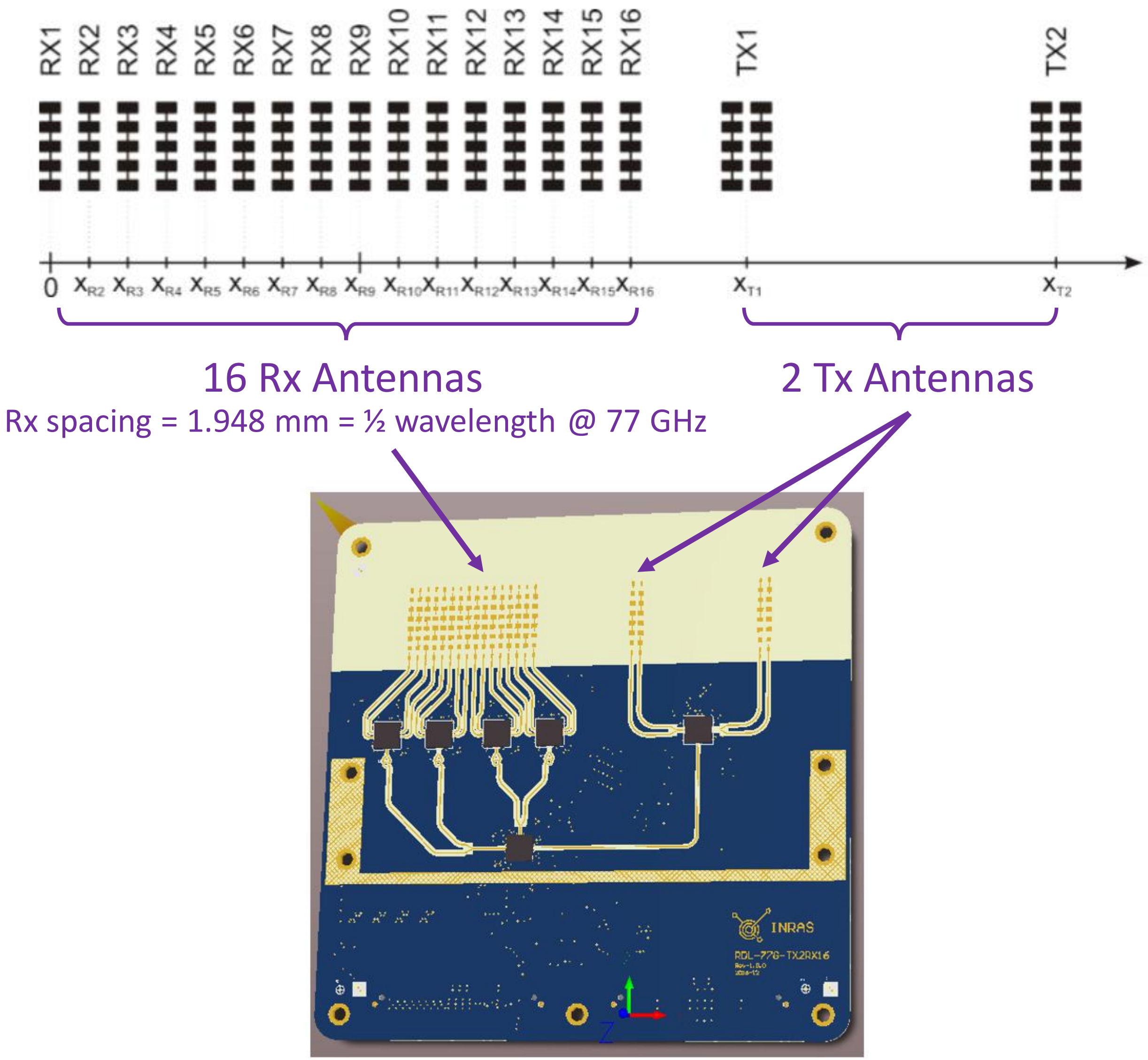
- GPU Solver supports multi-bounce
- Models complex scene interactions
  - Scene-object
  - Object-object
  - Object-scene-object
- Captures physics of radar's beyond line-of-sight sensing



MIMO Radars

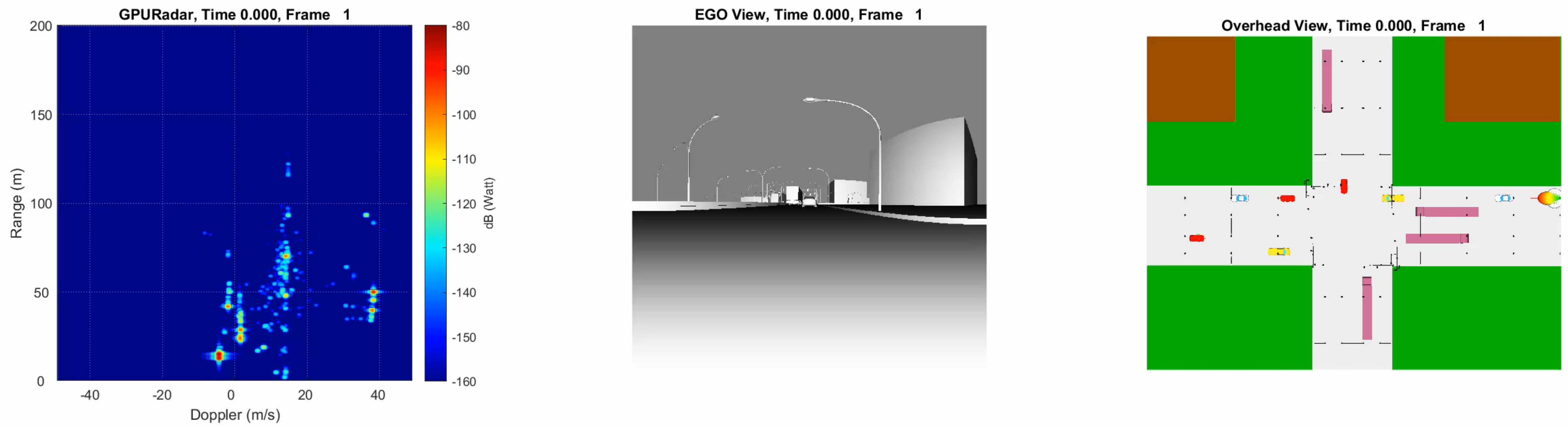
- Multiple-Input Multiple-Output (MIMO) radars are common
  - Linear arrays of multiple Tx and Rx
  - Interleaved modes
- Advantages
  - Improved imaging resolution
  - Detect angle of arrival
  - Electronic beam steering
  - More sophisticated signal processing
- GPU solver supports multi-Rx
  - Multi-Tx is coming soon

INRAS RDL-77G-TX2RX16 MIMO Radar



GPU Radar Project: Multi-Bounce Demo

Scenario: 6.00 sec @ 30 fps (181 frames), Single Channel
Simulation Time: less than 2 sec → ~90 fps
Hardware: NVIDIA Quadro GV100



Ansys

Ongoing Work and Future Directions

- Optimization for MIMO radar
- Multi-Tx, Multi-GPU
- Rough Surface Ground Model
- Integration into multi-physics driving simulators
- Weather modeling
- Additional Radar Waveforms
- HPC and Cluster Optimization