FLINT
FLEXIBLE INTERCEPTOR

A REALISTIC MISSILE SIMULATION SOFTWARE
FOR COMMODITY GAMES.

JULY 1, 2012

WIC MODERN WARFARE MOD

www.wicmwmod.com

AGENDA

INTRODUCTION

- **FLINT provides fairly realistic simulation of missiles and rocket projectiles in games, without requiring an extensive engine support.**

- **Written in python, it was written for World in Conflict (WiC). Could potentially be ported to other games.**

- **WiC FLINT uses frame buffering with Lerp due to game engine limitations. Engines providing dedicated projectile control layer (i.e. ARMA) would be more ideal.**

- **FLINT was originally created to provide a realistic replacement of HomingShooter for WiC. Today, it does lot more.**

PRIMITIVE ROCKET SIMULATOR

- **FLINT provides primitive simulation of forces acting upon rocket, taking account for basic aerodynamics and constants including drag, gravity and wind.**

- **Many games already offer this today (though WiC does not) on their native homing rockets. FLINT doesn't offer anything new in this area, pretty basic stuff here.**

- **Rocket specific impulse, Mach Max, G-loading, booster, sustainer, drag factor and tracking rates are available for customization.**

PRIMITIVE ROCKET SIMULATOR (CONTINUED)

- **FLINT rockets fired from high altitude are subject to minimum drag, as opposed to low altitude launches.**
  - Missile's range and sustained speed are increased at high altitude.
  - If you want your missile to have extended range (fly faster and longer), raise altitude and launch from angels high.
- **Numerous FLINT missiles (in accordance with real-life specs) will pitch up and "loft" into a ballistic trajectory immediately after launch, in order to extend the missile's range by cruising at higher altitude.**
  - AIM-120C fired from long range is a good example.
  - AGM-114 Hellfire is another example.

An Apache fires the AGM-114R semi-active laser homing Hellfire missile. The missile is seen pitching up and lofting into higher altitude to extend its range.

LEAD-PURSUIT HOMING

- **In almost every game, missiles fly using "pure-pursuit" trajectory, commonly known as Tail-Chase Navigation (TCN).**

  - Under TCN, missile's nose is always pointed at the target, chasing after the target as it moves. Very simple to implement.

  - In practice, TCN homing can't really intercept anything that moves. Missile needs to accelerate at least 10 times faster than the target to overtake.

  - Most games compensate this by over-engineering the rocket simulation – giving extreme unrealistic turning and tracking rates to TCN missiles to make them hit a target that would be impossible in the real world.

- **Under FLINT, most missiles fly using a form of "lead-pursuit" trajectory, in accordance with their real-life specifications.**

  - Under a lead-pursuit trajectory, missile leads the target and flies to where it thinks the target is going, rather than where the target is currently at.

FLINT HOMING LOOP

- **FLINT Homing Loop is the life cycle of every FLINT missile in game. Every time a missile is fired, it runs an infinite loop of the following tasks until it either (a) hits the target or (b) dies.**

1. Propulsion and Movement
2. Inertial Measurement
3. Data-Link / Remote Communication
4. Homing Guidance
5. Warhead Detonation Parameters

INERTIAL MEASUREMENT UNIT

- **FLINT engine provides an Inertial Measurement Unit (IMU) function for each missile.**

  - FLINT IMU measures missile's velocity, orientation and gravitational forces (basically an accelerometer) as it moves around in the game world.

  - FLINT IMU can provide the necessary dead-reckoning and velocity data to enable inertial navigation (INS).

  - INS is heavily used by most missiles for extended range shots, especially during "Lock on after launch" or "fire and update" situations.

WICO_FLINT MOVER

- **WICO_FLINT Mover function uses frame-buffering to move the missile in the WiC EXGame world. FLINT Mover runs exclusively on the server side and positional updates are sent to clients over the network.**

    - Frame-buffering requires the game to run at minimum of 27 frames per second or higher.

    - When frame rate drops below 27 fps, Lerp is used to "guess" but physics become unreliable during sustained low frame rates.

    - Latency is a huge problem during multiplayer. Linear interpolation is used to fill in the blanks between network updates to compensate, but it's still problematic due to the limited ways in which FLINT could interact with WiC game engine.

        - Missile IMU also generates interpolation data to Lerp, to help better compensate for latency while taking into account for true missile velocity.

- **Definitely NOT the best way to implement, but given the very elementary limitations of WiC SDK, it does the job fine.**

- **Could use better implementation under different game environment with more open SDK.**

MISSILE DATA LINK (MDL)

- **The amazing art of FLINT is its extensive data-link capability, where the launching unit in game has remote communication with the missile. This opens many possibilities:**

  - Missile seekers have limited power and range to see and track their targets. A missile with huge range doesn't mean anything when its seeker can't see jack at such distance!

  - Using data-link, the launcher or a remote controller (i.e. radar) could "command guide" the missile at long distance, to bring it closer to the target (mid-course guidance). Once close enough, missile can acquire the target using its own seeker.

  - Command Guidance (where launcher is exclusively controlling the missile) is very easy to simulate here.

- **Multi-Target Capability:**

  - Several missiles with data-link can be guided to multiple targets simultaneously, increasing the firepower.

  - Track While Scan (TWS) mode is also provided, combining detection/tracking sensor routines with memory data structures. Helps service large amount of targets simultaneously.

MISSILE DATA LINK (MDL)
(CONTINUED..)

- **Inertial Navigation (INS) and Missile Data Link (MDL) go hand-in-hand.**

  - INS is in the driver's seat, using dead-reckoning principle to blindly fly the missile without actually seeing a target.

  - INS suffers from "drift", where small errors in the measurement of acceleration and position are integrated progressively into a larger error, resulting in the missile totally drifting off-course over time.

  - MDL provides "refresh" to INS. Every often, controlling host (such as radar) transmits a target & missile position update over MDL, clearing out the drifts developing inside INS and putting the missile back "on course."

    - Think of yourself as running blind – your head will remember the last scene your eyes saw, including the target you're chasing after. Your head guesstimates how far you've run using dead-reckoning principle, however you'll end up running off-course since you really don't know where the target is. Once in a while, you briefly open your eyes to see a target for a second, updating the "image" in your head so you can correct your course. This is exactly how MDL and INS work together to keep the missile on course.

An Apache launches a Hellfire missile using Lock On After Launch (LOAL), taking advantage of the missile's inertial navigation system.

GUIDANCE MODES

Missile guidance is the art of hitting a moving target with precision. FLINT provides simulation of the following real-world missile guidance methods:

a. Command Guidance

b. Semi-Active Homing

c. Active & Passive Homing

d. Retransmission Homing (Track-via-Missile)

COMMAND GUIDANCE

- **The classic method. Launching host or another controlling unit (i.e. radar) has exclusive control of the missile.**

- **The missile is guided entirely by remote command. Missile itself does not have a seeker – it cannot think or track by itself.**

- **If the command unit (radar, etc) cannot see the target, the missile cannot hit the target.**

  - Easy way to dodge command guided missile is to fly very low. Since radar has difficult time seeing targets at low altitudes and is blinded by terrain, you can trash the missile.

An SA-15 Gauntlet (9K331 Tor) is seen command-guiding a SAM to a hostile aircraft.

The SA-15 has two command-guidance uplink channels. It is able to simultaneously provide guidance commands for two outgoing missiles.

SEMI-ACTIVE HOMING

- **Semi-active homing is a step up from command guidance. The missile has a seeker where it can track the target by itself.**
  - Semi-active homing is more accurate than command guidance, because missile is naturally closer to the target than the launching unit. Since missile is closer to target, it can gain more accuracy and clarity in its tracking solution.

- **Semi-active homing simplifies the missile seeker's sensor logic. Instead of having to search and identify the targets on its own, the seeker only tracks an "illumination beam" provided by an illuminator unit.**
  - It can be disadvantageous for air-launched systems, because the aircraft has to constantly "paint" or "illuminate" the target until impact. Not very good when you have enemy missiles coming at you.

- **Comes in two variants:**
  - Semi-Active Radar Homing (SARH)
  - Semi-Active Laser Homing (SALH)

SEMI-ACTIVE HOMING
(..CONTINUED)

- **A good example of semi-active homing missile in game is the AGM-114R Hellfire missile employing Semi-Active Laser Homing (SALH).**

- **Helicopter typically provides target painting & illumination. Survivability becomes a big problem when enemy employs double-digit "counter-helicopter" SAMs, such as SA-22 Pantsir, SA-15 Tor, etc. If you turn around to dodge the SAM, your missile will lose illumination and fail to track.**

- **Alternatively, infantry player with JTAC soldier can provide illumination by lasing the desired target using offensive ability. In this scenario, air player simply fires a missile to general vicinity of where the JTAC soldier is. Simply fire the missile on the ground, don't even need to aim at a target.**

  - The missile will fly in Lock On After Launch (LOAL) mode and will search for illumination beam from the infantry.
  - By co-playing with infantry JTAC, air player enjoys the benefit of "fire and forget", since the task of target illumination is left to the infantry.
  - Also known as "man in the loop" guidance.

An AGM-114R SALH Hellfire missile is seen tracking the laser illumination provided by a nearby JTAC infantry, which is hiding in the woods.

ACTIVE & PASSIVE HOMING

- **Similar in concept to semi-active homing – missile has a seeker that can track the target by itself, but with one huge difference:**

    - Missile is no longer tied to an illuminator or any other remote unit to support the missile. It is completely on its own, trained to kill – "fire and forget"

    - The missile itself has a sensor (i.e. active radar seeker, or a passive electro-optical seeker) that can search and track a target.

    - The range at which the seeker could sense and track a target is rather limited (limited to radar seeker's power output, visual range for electro-optical seeker, etc). This naturally limits the missile's maximum range to shorter distance.

- **Most games implement Active & Passive Homing as the only form of guidance for homing missiles (except missiles there have unlimited sensing/tracking range – quite unrealistic).**

- **Also known as "Lock On Before Launch (LOBL)" or "Fire and Forget" mode.**

ACTIVE & PASSIVE HOMING (..CONTINUED)

- **The short acquisition range of active/passive seekers can be extended to long distance using Missile Data Link (MDL).**

  - Missile flies with its seeker turned off (flying blind), using inertial navigation toward the target zone.

  - Inertial drift is constantly corrected by a command unit (i.e. radar), using mid-course updates transmitted over MDL.

  - Once the missile approaches close enough for its own seeker to track the target, it drops the MDL and goes "pitbull" – the missile from this point is now completely on its own.

- **Also known as "Lock On After Launch (LOAL)" or "Fire and Update" mode.**

An AGM-65 Maverick missile using Electro-Optical (EO) passive seeker is seen sharply turning to align itself toward its target.

RETRANSMISSION HOMING

- **Also known as "Track via Missile" or 'TVM' – became popular knowledge in 1991 when Patriot missiles using TVM were first employed in combat.**

- **Retrans. homing or 'TVM' is a hybrid combination of Command Guidance and Semi-Active or Active/Passive homing logics. The sensors from ground command unit are fused with the missile's own onboard sensor.**

- **Missile relays its seeker's target track/sensory data to the ground radar, which then fuses the track with its own sensory data. The ground unit then computes and relays guidance command back to the missile.**

  - Semi-active radar homing on steroids – combines the strengths of command guidance and semi-active homing together.

  - 2N redundancy homing – very difficult to jam. You have two separate units (missile and the ground radar) that are redundantly tracking you at the same time. If one is jammed, the other probably isn't.

HOMING LAWS

Homing law determines how the missile tracks and guides itself to its target as it sees it.   The following homing mechanisms are supported by the FLINT engine for guiding missiles in game:

- Tail-Chase Navigation (Pure-Pursuit)
- Proportional Navigation
- Predictive Guidance
- Bang-Bang Guidance
- Command to LOS (CLOS) Guidance

LINE-OF-SIGHT (LOS)

- **The basis of homing missiles, both in game environment and in the real world, heavily depends on the Line of Sight (LOS).**

- **The Line of Sight or "sightline" is a straight line from the missile seeker (pursuer's eyes) to the target. Simply speaking, LOS is the vector between the target position and the missile.**

- **As the target moves within missile seeker's field of vision, the angle of sightline also changes (the imaginary line between you and the target also moves as target moves).**
  - The rate of angular change in sightline is called LOS Rotation Rate.

TAIL-CHASE NAVIGATION (PURE-PURSUIT)

Also known as pure-pursuit, Tail-Chase Navigation (TCN) is the simplest form of guiding a homing rocket, where:

Acceleration = LOS

- **The missile is always pointed at the target, chasing after it. This produces a giant curvature based flight trajectory, commonly seen in CGI movies and games.**

- **Acceleration load required from the missile is very high near the end of its flight, as sharpest curvature of missile flight occurs at very end, since it has been chasing *after* its target the whole time. Simple evasive maneuver by the target will overload the missile's aerodynamic capability.**

- **Missile typically needs to accelerate at least 10 times faster than the target to overtake it with ease.** (impractical in the real world – a missile chasing after a fighter jet needs to accelerate to Mach 10+, which is not feasible)

  - TCN is used by almost every game today. Because TCN is so inefficient, game developers have to "cheat" in order to make it work, such as giving unrealistically agile turning rate, constant speed (unlimited rocket fuel) and extreme field of view to the missile, all of which are unheard of in the real world.

  - In WiC and Call of Duty, you can see homing missiles miss a target, then do a 360 degrees turn-around to go at it again, and the missile flies at a constant speed without any loss of energy. This never happens in real world.

PROPORTIONAL NAVIGATION

Also known as Collision Homing or a form of 'lead-pursuit', Proportional Navigation (PN) is simple to implement and very effective, where:

Acceleration = Closing Velocity * LOS Rotation Rate * NC

NC = Navigation Constant (typically 3 to 5)

- **PN forces the missile to "lead" its target, without having to know anything about the target's speed, or range to target.**
    - Very effective and simple to implement on any missile, even on heat-seeking missiles that don't have a radar. Missile always leads the target no matter what.

    - PN means that as you are chasing your target, your running-path rotation rate is faster than the LOS rotation rate by a constant multiplier – the Navigation Constant (NC).
        - Higher the NC, the faster you correct your path early in the flight, and less corrections are needed near the end.

    - Under PN guidance, the missile and the target appear to be on a mid-air collision course, where both objects appear to look "frozen" in state as they merge closer.

PROPORTIONAL NAVIGATION (..CONTINUED)

Simple PN as described earlier assumes constant air speed. Under maneuvering or accelerating situations, lead is ineffective.

- Even against non-maneuvering target, a non-maneuvering target still has an upward sensible acceleration of 1G.

To produce a more efficient intercept trajectory, the PN equation in FLINT engine is modified by an additional term:

$$Acceleration = Closing\ Velocity * LOS\ Rotation\ Rate * NC + (\ NC\ /\ 2\ ) * a_t$$

- This is also known as Augmented Proportional Navigation (APN).

- Under APN, the missile violently wiggles to align itself on the LOS to target, immediately after launch. Corrections in flight are minimal as it approaches closer to target. The missile collides onto the target in a near straight line.

A FIM-92 Stinger missile is seen tracking an aircraft target using Augmented Proportional Navigation (APN). The missile and target appear to be on a collision course. Missile flies nearly in straight line as it approaches the target.

PREDICTIVE GUIDANCE

Predictive Guidance (PG) can be argued as an advanced derivative of Proportional Navigation (PN), typically used by radar-guided missiles (i.e. AIM-120C, AGM-114L, MIM-104, PAC-3, etc).

- Real world versions of these missiles use similar in concept, but much more advanced forms of PG, though actual details are classified.
- In FLINT game world, we use a simpler elementary form of PG that is publicly known, but very effective for game presentation.

Proportional Navigation (PN) works without knowing the range to target, nor any kinematics data. Predictive Guidance augments PN by taking kinematics parameters and range into consideration.

Zero Effort Miss (ZEM)

In deriving homing guidance laws, it's useful to know what the miss distance would be if the missile and the target did nothing but continue on their current courses. This is called Zero Effort Miss or ZEM.

The logic behind ZEM is that if we know how far a missile will miss its target without effort, an acceleration command can be applied to reduce that miss. A homing guidance law can then be deriving using ZEM by repeating the following process:

1. Find ZEM
2. Compute missile acceleration to reduce ZEM
3. Repeat until ZEM becomes zero.

PREDICTIVE GUIDANCE (..CONTINUED)

Zero Effort Miss (ZEM) and LOS Rotation Rate

ZEM is closely related to the LOS Rotation Rate (LOS_r), which we used earlier for Proportional Navigation guidance.

ZEM's relationship to LOS rate (LOS_r) can be presented as follows:

$$LOS_r = ZEM / (LOS * T_{go}^2)$$

Alternatively,

$$ZEM = LOS * T_{go}^2 * LOS_r$$

- LOS Rotation Rate is easy to derive, as missile seeker itself can measure the change in angular rate of sightline (as all PN-guided missiles do).

- T_{go} represents **Time to Go** (estimated time to intercept), which can be calculated from the missile's radar range to target and its velocity measured by IMU.

PREDICTIVE GUIDANCE (..CONTINUED)

Knowing the ZEM and Time to Go, which accounts for missile's range to the target, the acceleration command should be proportional to the Zero Effort Miss and inversely proportional to the square of Time to Go until intercept as:

$$Acceleration = NC * ZEM / T_{go}^2 + (NC / 2) * a_t$$

- Predictive Guidance (PG) produces one of the most efficient collision-lead intercept trajectory, by combining the concept of Proportional Navigation (PN) and LOS rotation rate, with engagement geometry and kinematics data (Time to Go).

- Time to Go estimation can be augmented with various forms as seen fit, using range to target and kinematics parameters.

- Missiles using PG produce a straight-line intercept flight path, where the target appears to be being smacked by a meteor.

FLINT missiles employing Predictive Guidance in game:
- AIM-120C-7 (AMRAAM)
- AGM-114L-3 (Longbow Hellfire)
- Numerous advanced radar guided SAMs such as SA-15, MIM-104 (soon), etc.

An AIM-120C missile in game is about to home true to its target using Predictive Guidance (PG). PG interception resembles the look of a triangle being completed, with missile impacting like a meteor coming at a straight line.

BANG BANG GUIDANCE

The concept of Bang Bang is binary (0 and 1), "black and white" or "all or nothing" guidance, where maximum possible acceleration is applied ("banging") to the direction of the LOS rotation rate.

- FLINT missiles with side thrusters (ACM) or thrust-vectoring control (TVC) such as PAC-3 and AIM-9X combine bang bang with other forms of guidance. Bang bang is used to quickly align the missile on a desired intercept route, then guidance is taken over by PG or PN.
- Some Laser Guided Bombs (i.e. GBU-12) used Bang Bang guidance, where bomb's fins deflected fully rather than proportionally. This often resulted in over-corrections where the bomb keeps on deflecting back and forth to keep itself on the laser spot.
- Modern smart bombs (i.e. EGBU) use PN instead.

Bang Bang is expressed as:

*Acceleration = a_m * sgn(LOS * LOS Rotation Rate)*

a_m = Maximum applicable lateral acceleration of the missile

Bang bang is great at quickly aligning the missile to a desired intercept vector, but it also amplifies the noise in LOS rate, making it undesirable for sustained mode of guidance. It can result in constant over-corrections, wasting rocket energy early on in the flight.

COMMAND TO LINE-OF-SIGHT (CLOS) GUIDANCE

Command to LOS (CLOS) guidance keeps the missile in the LOS between the launch point and the target. It is a form of Command Guidance and is widely used by 1950's first generation US SAMs, present day Russian short-range radar-guided SAMs and anti-tank guided missiles (ATGMs).

- A commonly used form of CLOS is called SACLOS (Semi-Automatic Command to Light of Sight), which is more typically used on anti-tank guided missiles (ATGMs) and visual range SAMs.
- CLOS is effectively Beam-rider Guidance in concept, except for the method of implementation.

The distance between the missile and the LOS from the launcher (beam) is known as Cross Range Error (CRE). The Guidance Command Unit (GCU) at the launcher will compute and provide necessary acceleration command to bring the CRE to zero:

$Acceleration = K.CRE + a_{mp}$

K = Guidance Gain

a_{mp} = Beam motion (launcher LOS acceleration) term

The amount of error in CRE at the point of intercept is dependent on the range from launch point. Longer the distance, more error is introduced to CRE. This inherently limits the CLOS to short-range uses.

An SA-22 Greyhound (57E6-E Pantsir S1) missile is seen streaking across the sky and homing to its target using Command to Line-of-Sight (CLOS) guidance from the launch vehicle.

END OF PRESENTATION