AVC Specs

AVC Version 1.95

Document Version 1.21

1. Global Characteristics

- Portable C++ library – no dependency upon the OS and/or the CPU (successfully recompiled on Windows, MacOS X and Linux, on both 32/64bit architectures and on both little and big endian CPUs)
- Integration Frequency: 500Hz hard-coded (can be changed only by recompiling the whole library)
- multi-body for holonomic constraints; full multi-body support for generic constraints (such as articulated vehicles) can be added but requires additional work, as of this writing
- full interaction with terrain, external fluid (aerodynamics) and body to body collisions (with early reject tests based on bounding spheres); initial support already prepared for slipstream calculations
- custom integration algorithms based on modified verlet/leapfrog. All the equations, as of this writing, are explicitly integrated in a single pass (no iterative solver required)
- initial support for trajectory/internal state reconstruction/prediction based on saved “key frames”, for recording/replay and/or real-time networking applications (additional work is required to add full support).

2. Vehicle Model

A Vehicle in AVC is a collection of different, specialized components (C++ classes) that are designed in analogy to their real-life counterparts (engines, tyres, brakes, wings, suspensions and so on). A vehicle is built and allocated in real-time from a set of configuration files that allow a very high degree of freedom in the design of the virtual vehicle, without the need to recompile the application:

- the type and number of components a vehicle is made of is defined in config files and assembled at runtime, no real limitations are hard-coded in the library; for instance, a car can be designed with any number of wheels, differentials, wings, suspensions and so on
- each component can be named and has its own config file, thus allowing to build a library of pre-built “pieces” that can be assembled to create different vehicles; for instance, a set of different tyres can be designed, and replaced in the vehicle main config file by just changing the tyre names.

Of course, it is possible to create new components, creating new C++ classes from scratch or subclassed from the original ones, by replacing the factory constructor methods in the vehicle class; that allows to implements and integrate new components/algorithms into the AVC library with a very high degree of customization.

3. Component library

As of this writing, AVC supports and implements the following components:
- tyre (ruota_va)
- differential (differenziale_va)
- gearbox (scatola_cambio_va)
- steering system (scatola_sterzo_va)
- brake disk and caliper (freno_ruota_va)
- brake pump (pompa_freno_va)
- handbrake (freno_stazionamento_va)
- suspension (sospensione_va)
- sway bar (barra_rollio_va)
- third element/spring/damper (sospensione_centrale_va)
- (thermal) engine (motore_va)
- initial support for electric engine/generator (kers_va)*
- car bodywork (carenatura_va)
- wing (alettone_va)
- tank (serbatoio_va)
- ballast (zavorra_va)
- electronic controls (such as ABS or TCS)*

* additional work is required to expand those components to actually implement real-life counterparts

4. Masses and ballast

AVC recalculates in real-time the static mass properties of the vehicle. The basic requirements are the definition of the static unladen mass and the inertia tensor (in fact only the three main moments of inertia and the yaw-roll product of inertia are required); but it is possible to add components representing additional masses, such as tanks or (moveable) ballast. In that case AVC rebuilds in real-time the actual properties of the vehicle, depending on such masses and their positions.

5. Tyre model

AVC features three different tyre models:

- a simplified model based on three-dimensional stiffness characteristics (coefficients) of tyre carcass and contact patch, allowing to calculate how the tyre deflects/deforms in real-time in order to simulate both transient and steady state (combined longitudinal/lateral) conditions. This model supersedes the previous QFE tyre (Quasi-Finite Element) model, as it gives the same results in closed form without the need of iterative solutions of the internal solver. Friction of the contact patch is calculated using a brush-like approach, but adding the possibility to specify some time constants in order to simulate hysteresis in the grip/sliding transients even in this simplified model*
- an improved thermo-mechanical friction model that reuses some of the coefficients for the transient behaviour of carcass/contact patch deformations, but adds a whole new set of coefficients for the contact patch friction that is now calculated using the visco-elastic characteristics of the polymer and their interaction with the road (micro)asperities. The polymer characteristics are temperature-dependant and modified in real-time by a thermal model of the tyre that calculates, at each time step, how the heat flux is exchanged between the tyre, the road and the surrounding air
- a sophisticated terra-mechanics friction model, based on loose terrain (gravel, snow) characteristics (soil adhesion and angle of friction) and tyre-soil interaction (sinking, shear-stress length) in order to simulate off-road conditions with a high degree of realism.

AVC tyres, also, feature a simplified model for surface soil “dirt” and its interaction with tyre friction (for instance, hydroplaning due to the presence of a film of water on a hard surface).

As with any other AVC component, tyres can be subclassed and extended/expanded in order to implement custom tyre models.

* this model is no more available, although it can be retro-fitted, as the thermo-mechanical one proved to be far superior and has superseded the simplified one.

6. Suspensions and steering

Tyres can be attached to suspensions and/or steering assembly, so that a vehicle can have any number of wheels independently suspended and/or steered (allowing to simulate 4-steering wheels system, for instance).

A suspension in AVC is defined by a config file that describe tyre motion vs suspension travel, or, in other words, how the tyre is displaced when the suspension is compressed/rebounds. AVC suspensions are a generalization of “camber curves” that extends the concept to toe-in/out and curves that define how the tyre moves, in the car reference frame (track and wheelbase change), with suspension motion. This approach allows to simulate, with a very high degree of realism, a wide range of types (double wishbone, McPherson, twist-beam, just for example) without any need to define the actual suspension scheme, because it is enough to define what the suspension “does”, instead of how it is “made”: AVC is able to internally calculate how such a motion influences in real-time roll-centers and/or anti-dive or anti-squat effects, along with track or wheelbase dynamic changes.

Suspensions can define static and dynamic (where applicable):

- camber
- toe-in/out (including bump-steering)
- caster
- king-pin inclination
- scrub radius

Suspensions can be coupled with standard (passive) devices such as:

- linear and progressive springs (with or without precharge)
- helper springs
- packers
- 2-ways and 4-ways dampers

Pair of suspensions can also be linked/coupled by means of “third elements” such as:

- sway bars
- third springs
- third dampers
- j-damper systems

A steering system can be applied to any wheel of the vehicle: in that case, apart from the geometry already defined in the suspension, additional parameters can be specified, such as the steering ratio/decoupling, the steering non-linearity, and the ackermann effect.

7. Powertrain

AVC features an innovative algorithm, based on an electric circuit analogy, that allows to build very complex powertrains by simply assembling in the right order the right components. AVC does not make a distinction between a simple go-kart or a complex AWD eight-wheeler, the solver uses the same algorithm in both cases, calculating how the torque applied to each component interacts with the other components with the constraint of angular motion conservation.

An Engine is usually the root node of the powertrain, but additional engines can be added at any stage in the network. AVC thermal engines are defined by the torque curve, the boost curve (for turbocharged units), and other properties (redline, inertia, gas and turbo lag, consumption and so on) that represent the dynamic behaviour of the unit. AVC has an initial support for kers, too, although a full implementation of electric engines is possible but requires additional work.

Engines can either by directly coupled to the powertrain network (go-kart configurations) or decoupled using components such as clutches/torque converters. Gearboxes can be added to change the final speed/torque ratio; AVC features both manual and automatic gearboxes, and initial support is already provided for CVT systems (although additional work is required for a full implementation).

Last but not least, the differential class (differenziale_va) can be used, just like its real-life counterpart, to split motion between two axles, and by combining and linking the right number of differentials together any number of wheels can be driven by a single engine.
Moreover, AVC differentials support:

- open or (passive) LSD differentials, where the LSD action can be controlled by reaction torques, relative angular speed (slip), or both, thus providing support for the main types of differential (salisbury, torsen, viscous couplings)
- static torque bias split, in order to simulate central differentials; the actual torque split can dynamically change in real-time if an LSD differential is used
- different axle gear ratio
- different efficiency ratio.

As with any other AVC component, powertrain objects can be subclassed and extended/expanded in order to implement custom types, such as active (electronic) differentials.

8. Brake systems

AVC allows to simulate both braking systems usually found on cars, the main braking system and the parking brake. The braking system is split into two component, the actuator (the brake pumps connected to the brake pedal or the handbrake) and the actual brake mounted on the wheel. For each component it is possible to define its characteristics, such as:

- the front and rear pump dimensions
- the front and rear braking balance (adjustable in real-time, if applicable)
- the brake dimensions, masses and inertias
- the thermodynamic model of the brake (thermal capacity, air cooling and so on)
- the thermal characteristics and efficiency of brakes and brake pads

9. Wings and aerodynamics

AVC supports the interaction of a vehicle with the external fluid (air) allowing to define components that control aerodynamics of the main car bodywork (carenatura_va) and its additional wings (alettone_va). For each aerodynamic component it is possible to define:

- a first set of coefficients controlling the basic characteristics of the aerodynamic component, such as the resulting force and its pressure center when the component interacts with an air flux; such coefficients define the three main cross sections of the component along the three main axis
- a second set of coefficients controlling how the basic characteristics change with air speed, allowing to design linear, parabolic or even higher-order curves that control how drag and lift increase with speed; lift and drag speed curves can have different shapes (coefficients)
- a third, optional set of coefficients controlling how the previous coefficients change in real-time by changing the car height (aeromaps); multi-dimensional aeromaps are supported, as the number of “height control points” are user-definable (for instance, it is possible to define a simple bi-dimensional aeromap controlled by the car front and rear height, or a complex 4-dimensional one using the height of the four car corners); moreover, each component can have its own aeromap, and the aeromap can include additional dimensions for adjustable parts in order to simulate what happens when the aerodynamic setup is changed (for instance, a change in the wing angle of attack) even in real-time

- an additional, optional set of coefficients can define any “aero-boost” possibility of the component, in order to simulate advanced wings designed with features such as induced/forced stalling (f-ducts) or DRS capabilities.

AVC aerodynamics is not simply “static”, the external fluid (air) can have its own speed (wind), temperature, and density in order to simulate a wide range of situations. AVC aerodynamics has initial support, too, for slipstream calculation, although additional work is required if full interaction with other aerodynamic shapes moving around the vehicle is needed.

10. Electronics

AVC allows to define electric/electronic devices that can modify the input to vehicle components in order to assist the driver, to control the vehicle stability, or to change the torque sent to the wheels. Such devices can be easily integrated inside the network of the other components so that they can interact with the input/output of the physical parts of the car. As of this writing an initial implementation of anti-lock brake system (ABS), traction control system (TCS), and kinetic energy recovery system (KERS) has already been done, but those system can be expanded to represent more complex real-life ones and many other electronic devices can be easily added and integrated in a seamless way into the library using the same framework and philosophy.