

A kind of crane loading and unloading simulation based on finite-state Machine

Guang Hong^{1, a}, Weibing Bai^{1, b}, Xin Yu^{2, c}

¹Department 5, Wuhan Mechanical Technology College, Wuhan, 430075, China

²Department 1, Wuhan Mechanical Technology College, Wuhan, 430075, China

^aemail:xinye5804@sohu.com, ^bemail: 37604287@qq.com, ^cemail: yx2150@126.com

Keywords: finite-state Machine; crane loading and unloading; virtual simulation; Unity3D engine

Abstract. Aiming at the insufficiency of the actual crane, finite state machine technology was adopted for setting up an emulational training system. The system implemented a kind of crane loading and unloading operation. Therefor a sort of finite-state machine for the crane was set up. By the dynamics analysis for the crane, several states of the finite state machine were established during loading and unloading process. All the states were bewritten with some mathematic models. The models were simplified to adapt to the simulation system. Then the flow for establishing finite state machine was given, and the concrete methods in Unity3D engine for establishing finite state machine were described. Finally an instance of a kind of crane loading and unloading training system was implemented based on the aforementioned models and methods. The instance shows that the models are accurate and the finite-state machine states are logical.

Introduction

The crane loading and unloading training requires adequate real equipment, especially for large numbers of students. Otherwise each student has not plentiful training time and the training efficiency is awful. So a simulation training system is necessary for enhancing the training efficiency.

The virtual crane system is essentially a kind of mechanical simulation. Usually there are three kinds of methods for mechanical simulation.

One is such professional mechanical simulation tool as ADAMS, Solidworks, Pro/E and so on. These tools have adequate precision for analyzing force and movement, but they are hard to develop and have worse interaction. So these tools are not suitable for our application.

Another is virtual reality modeling language (VRML).VRML has good interaction and can be distributed as an internet application. But VRML has less mechanical processing ability. Analyzing force and movement is much hard.

The third is VR engine or game engine such as Virtools, Open Scene Graph [1], Vega [2], and Unity3D etc. These tools have much better interaction and have embedded physical engine to analyzing force and movement. So these tools can simulate many mechanical phenomenon and physical phenomenon. And the precision is adequate for the training simulation. Furthermore the application is easy to transplant to internet for remote training.

The crane training system is a teaching and training system. It requires good precision for analyzing force and movement and better interaction for native and remote operation. So VR engine or game engine is the best tool for the crane training system. Unity3D is final choice for its cross-platform and convenience.

Finite state machine (FSM)[3,4,5] is a mathematic model describing finite state, transition behavior between all the states and all the actions of the states. FSM can be shown by a directional figure, it can be defined as:

$$M=(S, \Sigma, \delta, S_0, S_F) \quad (1)$$

S ——a set of the finite state
 Σ ——a set of all the events

$\delta: S \times \Sigma \rightarrow S$, transition function, commonly it defined as $\delta(S_1, a) = S_2$, that means if event a happens, the state will transform S_1 into S_2 .

S_0 is the subset of set S . It is a particular state, namely is initial state.

S_F is the subset of set S . It is also a particular state, namely is final state.

The crane loading and unloading state analysis

The crane is divided into four main parts that are gyre one, lift one, range one and vice crane arm one. It is shown in Figure 1. all the joints among the parts are fix joint or hinge joint. The Fixed joint groups together 2 rigid bodies, making them stick together in their bound position. The Hinge Joint groups together two rigid bodies, constraining them to move like they are connected by a hinge. It is perfect for doors, but can also be used to model chains, pendulums, etc. The gyre framework can rotate 360° and the lift framework can translate freely along its track. The whole arm can rotate by range framework translating. The vice arm can rotate along its axis.

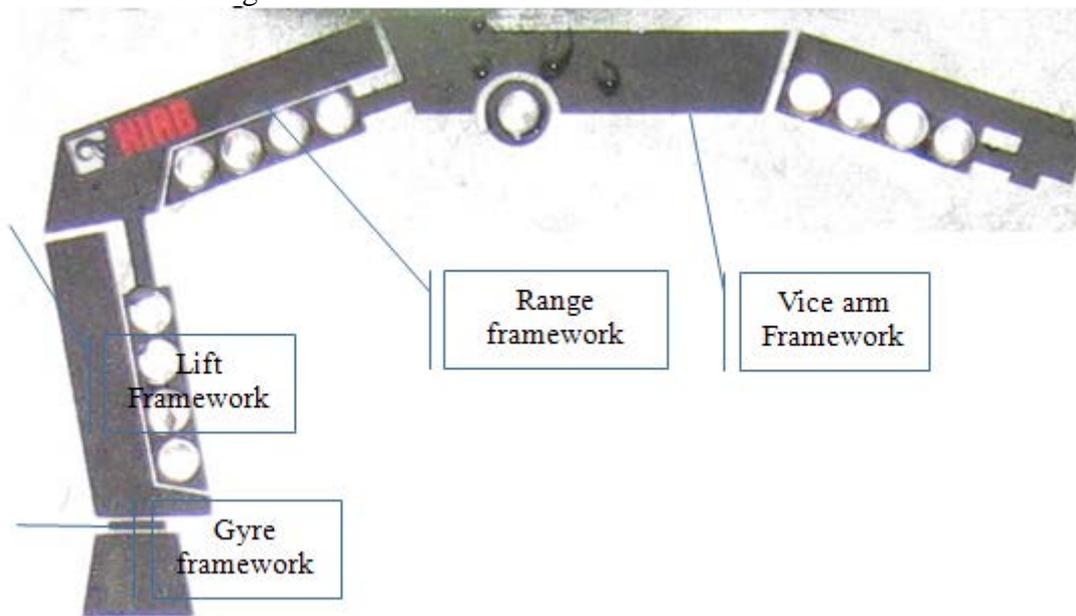


Fig.1 the crane structure

According to analyzing the crane structure, the crane loading and unloading can be divided into six states. S_0 is the initial state, S_1 is the lift state, S_2 is the gyre framework rotating state, S_3 is the range framework translating state, S_4 is the vice arm rotating state and S_5 is the final state. All the states need individual mathematic model.

The uniform coordinate is absolutely necessarily for establishing the models. There are two kinds of coordinate. One is the world coordinate; the other is the local coordinate. The world coordinate has only one origin; however the local coordinate has individual origin. The local origin can be transform into the world origin. If an object's origin is at the position $v_0(x_0, y_0, z_0)$ and the object is at the pose $v_1(x_1, y_1, z_1)$ in the world coordinate then any vector v in the local coordinate can be calculated as follows:

$$wv = v_0 + v \tag{2}$$

$$wrv = v_1 + v \tag{3}$$

wv means the position in the world coordinate;

wrv means the pose in the world coordinate;

Rotation transform is more complex. Define a quaternion $q, q = \lambda + P_1i + P_2j + P_3k$

If an object rotates from v to v' then $v' = q^{-1}vq$

q is a quaternion $q, q = \lambda + P_1i + P_2j + P_3k, i, j, k$ are all imaginary number unit.

then v' in the world coordinate is wrv_{roll} ,

$$(4) \quad wrv_{roll} = q^{-1}vq + q^{-1}v_1q$$

All the states are the same modeling method. For instance, State one includes 3 periods. When the lift joystick is enabled, at first the force is not enough to lift the heavy object. Then the lift framework rises up when the force is enough at an invariable rate, and the heavy object is along with the same rate raises up. Finally the heavy object stops lifting when the joystick is disenabled.

Suppose x denotes the lift displacement in local coordinate, and m denotes the quality of the heavy object, and F denotes the inner tensile force.

The heavy object are under gravity, tensile force and the pull force. When F is less than gravity, the object is immovability.

When F is little bigger than gravity, the object starts to lift. The span from $F=0$ to $F=mg$ is t_0 .

$$\int k_0 t = mg \quad (5)$$

The parameter k_0 is a constant and is confirmed by measure from real crane. The parameter t_0 is confirmed from formula 5.

When the lift framework lifts, the lift displacement is direct ratio motion time.

$$x = k_1 t \quad (6)$$

The parameter k_1 is a constant and is confirmed by measure from real crane.

When the lift joystick is disenabled, the object begins to stop, the lift displacement continues to increase with a decreasing amplitude.

$$dx(t)/dt = k_2 \exp(-t/\lambda_0) \quad (7)$$

The parameter k_2 and λ_0 are both constant and are confirmed by measure from real crane. In theory the object stop moving when t is infinite. In fact if t equals to $5\lambda_0$, then $\exp(-5) = 0.007$, approximately the x equals to 0.

State 2, state 3 and state 4 have the same modeling method as state 1.

State 5 is the final state. It means the heavy object is closer enough to a certain position.

$$d^2 = |v - v_{PF}|^2 + |v_R - v_{RF}|^2, d \leq \varepsilon \quad (8)$$

v_{PF} means a certain position vector, and v_{RF} means a certain pose vector, v means the real position vector, v_R means the real pose vector, ε means a little enough constant.

The FSM model as follows:

$$M = (S, \Sigma, \delta, S_0, S_5)$$

$\Sigma = \{\text{lift joystick is enabled and } t=t_0, \text{ lift joystick is disenabled and } t=5\lambda_0, \text{ gyre joystick is enabled and } t=t_1, \text{ gyre joystick is disenabled and } t=5\lambda_1, \text{ range joystick is enabled and } t=t_2, \text{ range joystick is disenabled and } t=5\lambda_2, \text{ vice arm joystick is enabled and } t=t_3, \text{ vice arm joystick is disenabled and } t=5\lambda_3, d \leq \varepsilon\}$

$$S = \{S_0, S_1, S_2, S_3, S_4, S_5\}$$

$$\delta(S_0, \{\text{lift joystick is enabled and } t=t_0\}) = S_1$$

$$\delta(S_1, \{\text{lift joystick is disenabled and } t=5\lambda_0\}) = S_0$$

$$\delta(S_0, \{\text{gyre joystick is enabled and } t=t_1\}) = S_2$$

$$\delta(S_2, \{\text{gyre joystick is disenabled and } t=5\lambda_1\}) = S_0$$

$$\delta(S_0, \{\text{range joystick is enabled and } t=t_2\}) = S_3$$

$$\delta(S_3, \{\text{range joystick is disenabled and } t=5\lambda_2\}) = S_0$$

$$\delta(S_0, \{\text{vice arm joystick is enabled and } t=t_3\}) = S_4$$

$$\delta(S_4, \{\text{vice arm joystick is disenabled and } t=5\lambda_3\}) = S_0$$

$$\delta(S_0, d \leq \varepsilon) = S_5$$

The course of crane loading and unloading is obviously discrete, so the FSM is suitable. The flow of the FSM is shown in Figure 2.

Implement the Simulation training system

All parts of the crane are rigid bodies. There are only translation and rotation action for each part. Different parts are connected by joint in Unity3D engine. There are four kinds of joint, Fixed Joint, Hinge Joint, Character Joint and Configurable Joint. The configurable joint is an extremely flexible joint giving us complete control over rotation and linear motion. It gives us control over motors,

drives and joint limits for each rotation axis and linear degree of freedom. So all the joints of the crane are configurable joints.

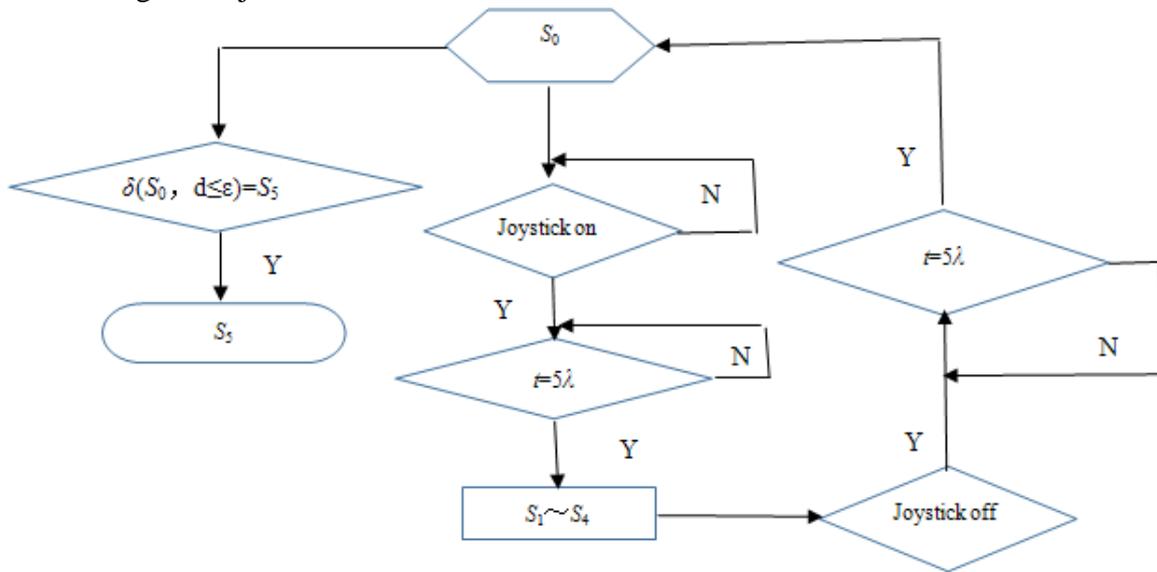


Fig 2 the flow of FSM

After configuring the parameter, not the driven motion but the active motion needs control. The driven motions are implemented by the Unity3D engine.

After configuring the rigid bodies and the joints, the crane has been provided with physical trait and motion parameter for the simulation. But the driving force has to manually configure. Unity supports keyboard, joystick and gamepad input.

A game joystick is substitute for the joysticks of the crane. When a joystick button is pressed, a force or torque will add to the object. The value and continuous time are determinate by formula 5 and formula 6. Then an event happens in FSM, the state of FSM may change into $S_1 \sim S_4$, and the relevant action does. When a joystick button is released, the force or torque will disappear from the object. The parameter is determinate by formula 7. Then an event happens in FSM, the state of FSM change into S_5 , and the S_5 action does.

Firstly initializes the FSM parameter such as states set, events set, transition function, initial state and final state. Then creates a whole cycle, the cycle updates per frame and inquiry if any event in FSM happen. If no event happens, waits. If some event happens, updates the state. If the state is final state then suspend FSM otherwise does action of the state and return. The flow of implementing FSM is shown in Figure 3.

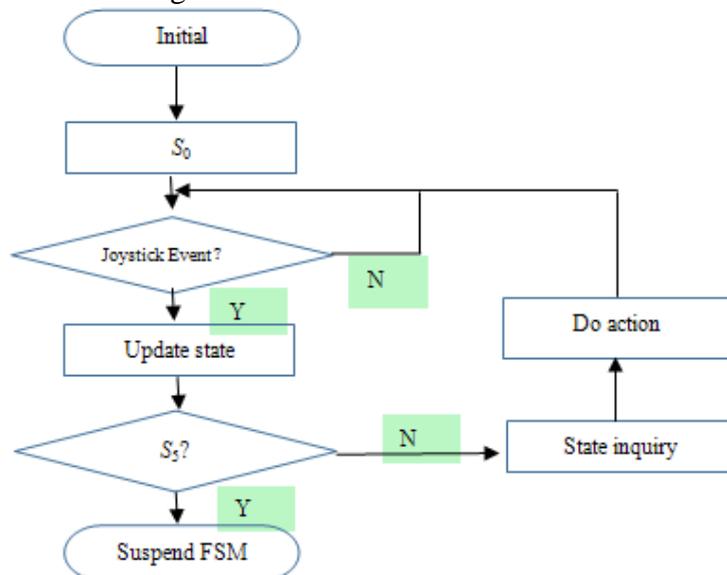


Fig 3 the flow of implementing FSM

Conclusion

In this paper, a kind of crane loading and unloading FSM is implemented. By the dynamics analysis for the crane, several states of the FSM were established during loading and unloading process. All the states were bewritten with some mathematic models. The models were simplified to adapt to the simulation system. Then the flow for establishing finite state machine was given, and the concrete methods in Unity3D engine for establishing finite state machine were described. Finally an instance of a kind of crane loading and unloading training system was implemented based on the aforementioned models and methods. The interface of the system is shown in fig 4. The instance shows that the models are accurate and the FSM are logical. The models are simply, intuitionistic and easy to implement. Farther work is design a real joystick the same as the real crane. Then the efficiency is better.

References

- [1] Gong Shui-qing,Wang Wei,Li Pei-lin,Tang Qian. Virtual Hoisting Training System and the Enabling Technologies [J]. Machinery Design&Manufacture, 2012 (11) 31-33
- [2] Zhang Li, Xu Yi-he,Wang Yi-jun,Zhang Wei-jin. Lifting training simulators technical scheme design [J]. Hoisting and Conveying Machinery, 2013 (2) 10-13
- [3]Black PE (2008) Finite state machine. Dictionary of Algorithms and Data Structures, ed Black PE (U.S. National Institute of Standards and Technology). Available at www.nist.gov/dads/HTML/finiteStateMachine.html.
- [4] Jun Yaoa, Hao Yana, Shamik Dasb, James F. Klemicb, James C. Ellenbogenb, and Charles M. Lieber. Nanowire nanocomputer as a finite-state machine [J]. Proceedings of the National Academy of Sciences of the United States of America,2013 111(7) 2431–2435
- [5] Daniel O. Sales, Diogo O. Correa , Leandro C. Fernandes , Denis F. Wolf , Fernando S. Osório. Adaptive finite state machine based visual autonomous navigation system [J]. Engineering Applications of Artificial Intelligence, 2014(29) 152–162